



December 1988

Vol. 2

Nº 3

Price £1.80

archive

The Subscription Magazine for Archimedes Users

ProArtisan Review

Dabs' Archimedes BASIC Compiler

Disc Recoverer

Shareware Takes off!! / Times Crosswords

Configuring the Archimedes / Music Keyboard

Thoughts of a PC Emulator User

BBC Software Compatibility List / LISP Review

Christmas comes but once a year...

Yes, but they don't half come round with amazing regularity! This is Archive's second Christmas and we've now had over 2,000 subscriptions. I'm going to try to get the January magazine finished before Christmas so I can have a break, so would contributors please note that I'd like articles, notes etc on 640k disc please, a.s.a.p. Ta!

I hope you all have a very Happy Christmas – a well-earned rest and some good re-creation in the best sense.

With very best wishes for Christmas and the New Year,



Thanks be to God for His inexpressible gift !

2 Corinthians chapter 9 verse 15.

In the Readers' Comments column there is a letter from Brian Oliver saying basically that, for a number of reasons, he was dissatisfied with Minerva's Gammablot. We sent a copy of his letter to Minerva who unfortunately did not manage to get a reply in to us until the very last minute before publication, so this is the only space left to print their reply! – not ideal, but better than nothing. To be fair to both parties, you should read Brian's criticisms first before reading Minerva's reply which is as follows...

"We are sorry that Mr Oliver was unable to get the best out of Gammablot but he appears to have

missed most of the main features, perhaps by not making full use of tutorial provided. We really feel that supporting a 256 colour mode was more important than mode 20 for the 3% of multi-sync owners.

"Gammablot is used widely to produce professional graphics – certainly more so than Euclid which seems a strange comparison for Mr Oliver to draw.

"Any of the Archive readers who are apprehensive after reading Mr Oliver's letter are invited to read the many excellent write-ups by professional, independent reviewers or to contact us to have their doubts dispelled."

Archive

Volume 2 • № 3 • December 1988

Contents

Hardware & Software Available ...	2	Small Ad's	25
Shareware Takes off!	3	Dabs' BASIC V Compiler	27
Help!! needed and offered	5	BASIC V Forum	30
Matters Arising	8	BBC Compatibility List	35
Readers' Comments	10	Knitware Systems Programming	36
Hints & Tips	12	LISP – A Wider View	37
First Word Plus Hints & Tips	17	Disc Recoverer.....	41
What price ProArtisan?	19	ASCII, Printer Drivers & FWP ...	44
Contact Box	21	Assembly Programming – 5	46
Quazer Review	21	Archimedes Music Keyboard	48
Times Crosswords	22	Thoughts of a PC Emulator User	51
Configuring the Archimedes	24	Fact-File	53

Hardware & Software Available

- **Repton 3** is here! Well, just about – it should be available before Christmas. We have the pre-release version of John Wallace's adaptation of the ever popular BBC celebrity. The Archimedes version, which runs in mode 9, has four different tunes and an 'off' option(!), it runs 25% faster than the BBC version and it has a very useful extra feature of allowing you to save the position you have reached so that if you have just negotiated a difficult bit, you don't have to go back to the beginning if you get killed. There is also an editor so that you can design your own screens (16 colours) and your own men. It's called Repton 3 because you've got three of the Repton stories in one: The Life of Repton, Around the World in 80 Screens and Repton through Time. All the passwords are different from the BBC ones as are the competition codes, so it's start from scratch time for all you BBC Reptonians. £19.95 from Superior Software. (£18 through Archive.)
- **Jet Fighter** – Minerva's latest games program £14.95 (£14 through Archive) – "Alone on a forgotten planet, you must defend yourself against all-comers whilst, aided by your jet powered backpack, you re-build your broken rocket ship, then fuel it and take off to the next level."
- **PacMania** – At last, the only game I ever spent time playing on the BBC – PacMan – has now been converted to the Archimedes. Grandslam's PacMania (£19.95 or £18 through Archive) is a 3D version where you can, amongst other things, jump over the monsters! (Needs 1 Mbyte minimum.)
- **Home Accounts** – £49.95 from Minerva (£46 through Archive) is their first application that will run under RISCOS in single or multi-tasking modes. Designed to keep club or home finances, it will control multiple bank accounts, credit cards or building societies and does automatic posting of standing orders and transfers of funds between accounts. It includes budgeting and cashflow forecasts, provides a full analysis and even warns of bank charges or overdraft situations. Also includes simple bank reconciliation. N.B. this is self-standing software and not an add-on to System Delta Plus as is the case with the Business Accounts software. (See small ads if you need a copy of Business Accounts.)
- **Clares' Alphabase** – This has rather got forgotten, but it is a perfectly respectable database for the Archimedes at £49.95 (£46 through Archive.) We have a copy for review if anyone is interested.
- **Software Developer's Toolbox** from Acorn contains a Symbolic Debugger and a set of utilities working in C, Pascal, Fortran 77 or BASIC: Common – lists the most common words in a list of files, Dbug – machine code debugger for use with multiple languages, Diff – compares two files and lists their differences, Grope – searches files for a given pattern, Libfile – manipulates library files, Link – links object files and libraries to form an image file, Amu – controls compilation of multiple sources in large programs, Memtest – tests the application memory of the machine, Objlib – lists the symbols in a library file, Shell – allows BASIC to be used as a command language, Squeeze – compresses image files to conserve disc space, Wc – counts characters, words and lines in a file. Acorn price £199 + VAT, Archive special price (while stocks last) £130
- **Nine discs of programs** – We've had a list of nine program discs at £5.99 each from David Pilling, P.O.Box 22, Thornton Cleveleys, Blackpool, FY5 1LR. (Buy four, get one free!) They are: EMACS programmers/text editor, Micro Spell, Fortune Cookie, XLISP, C Toolkit, Kermit, Chess, Cross Star, File Tools.
- **PD-PROLOG** – A public domain version of PROLOG is available from The Advisory Unit, Endymion Road, Hatfield, Herts, AL10 8AU. It works under the PC emulator, so just send them a formatted MS-DOS disc along with a return envelope and postage.
- **Extended Precision from within BASIC.** – EpBAS from Abacus Training allows you to use all 27 floating point unop & binops from within BASIC. You can get 19 significant figure precision in BASIC and use numbers from 10^{-4931} to 10^{+4931} . Extended precision numbers are stored in BASIC strings or Ep registers. EpBAS is £30

inclusive and comes with examples on disc e.g. 2 x 2 determinant, the square root of 2 as 2*SIN(ATN(1)) and the solution of a triangle.

- **Maths Co-Processor** – No, not available yet, but I thought you'd like an up-date. The latest target is 2nd quarter '89 and the price is still quoted as 'between £500 and £1000'.

- **ArVis Videographics Expansion** – Video Electronics Ltd have a range of products for the Archimedes: a PAL encoder board, a video Controller board and a very exciting looking Videographics podule. The podule extends the display capabilities of the Archimedes from 256 to 32000 colours and is complete with a built-in real-time colour digitiser and a 384 kbyte x 16 bit video framestore. It also combines the digitised video images with live video and the normal Archimedes graphics output for professional videographics applications. Video Electronics sent us some very impressive screen shots of digitised images.

- **Colour Converter improved** – Lindis International have added some extra software for their colour converter. This is in addition to the original software and registered users will automatically be up-graded. There is now "error correction dithering" which allows the simulation of 256,000 colours and a mode 12 facility to allow you to import digitised pictures into Artisan or First Word Plus. (The colour converter basically takes a colour signal from a recorder or a camera, splits it up into the three colour elements and allows you to feed each colour separately through to the Watford digitiser. The three digitised images can then be superimposed to give the colour picture. Matthew Treagus will be reviewing it in due course.)

- **Yet another art package** – Pax is a 256 colour art package from Z & Z Software, price £15.95.

- **Cobra** – a snake-eating-up-the-bits game for the Archimedes, £6 from Richard Millican 10 Stokesay, Bidston, Birkenhead, L43 7PU.

- **RISC OS** – next month we will be giving a lot more detail about the RISC OS 2.0, but you are now able to place advance orders for £33 inc. VAT + £3.00 p&p. (We can't say exactly how much it will be for overseas customers because we don't know how heavy it will be, but I estimate that you will

have to add £10, £12 or £15 postage, depending on the postal area, instead of £3.) If you want to place an advanced order, please send a separate post-dated cheque (April 1st 1989!) which we will hold until we have the RISC OS packages ready to send out to you. Remember that after an initial period, the price of RISCOS will revert to the full £56.35 + postage.

- **Euclid up-grade** now available – Euclid, the 3D graphics package has been up-graded and registered users will get a free up-grade. New features include lighting effects and the ability to colour objects by selecting from a range of colours in a colour wheel. The viewing of scenes from different angles has been simplified as have the methods of constructing scenes and there is a new help facility. (Euclid costs £45 from ACE computing or £42 from Archive.)

- **Wimp Template Editor** has been delayed a little but should be ready by the time you read this. Adrian has been making sure that it is compatible with RISC OS, rather than having to up-date it later.

Review Software Received...

Apart from reviews already being written we have received review copies of the following software: Alpha-Base from Clares (months ago someone offered to do a review though nothing came of it), Cobra from Richard Millican, Extended Precision in BASIC from Abacus, Floating Point Assembler from Abacus, Ovid Toolkit module, Pax, 256 colour art package from Z & Z Software, P.R.E.S. Archimedes Utilities.

Shareware Concept Takes off!

Since the early days of Archive, we've been intending to get going on setting up the Shareware concept but it's never really taken off until now. We did do the graphics demo disc fairly early on and then the first Shareware disc, both of which are valuable in their own way, but this latest Shareware disc really is in a different league. Last month, I wasn't exactly sure what we had available to put on it – all I knew was that there was quite a bit of it. When Adrian presented me with the finished disc, I was certain that it would be well worth £3 of anyone's money. Just look at the list below and you'll see what I mean. I'm not saying that all the programs are wonderfully well written, though some certainly

Hardware & Software Available

are, but if just one of the major programs is of use to you, it's well worth £3.

More of the same, please!

So what we want now is more of the same! It is in everyone's interests (apart from the commercial software houses – Sorry chaps!) to generate and distribute more of these Shareware discs. So if you have a program or programs that you have written that are at least reasonably good, send them in. Also, if you take one of the existing programs and add various new facilities to it or improve others, send it in and we can put the modified program on a later disc. If the program needs documentation, please send it as a readme file with the program.

Here is a list of what is on the discs that we currently have available.

Graphics Demo Disc – 49 graphics demo programs plus "MenuMaster" – a program which allows you to add in your own demos, and edit the order in which the programs are run, including repeating some demos and making the demo into a complete loop.

Shareware Disc № 1 – MenuMaster with 7 more graphics demo programs plus Life, Mandelbrots, European Geography, Structured Directory Lister.

Shareware Disc № 2 – DFS reader, backup and archiver. Nine more graphics demos, 256 colour Sprite Editor, CMOS ram Editor, Disc Copier, LQ printer Font Definer, Matrix Functions, Memory Mappings & Vector Listings, BASIC Fast Screenload, Connect Four, Mastermind, Solitaire and Star Trek.

Monthly Program Discs

We are still getting requests to organise a monthly program disc subscription. The position is, as I think we said in an earlier issue, that we will accept pre-payment for any number of program discs at £3 each, so you can pay for 3 at a time, 6 at a time, 10 at a time or pay for the next 5 years' discs all at one go if you really want to! I'm afraid we just do not have time to formalise it into a disc subscription and I think that at £3 (compared with A&B's £7, Acorn User's £50 for 12 and RISC User's £5.25) you are getting good value for money. **A**



4-SLOT AND 2-SLOT BACKPLANES FOR ARCHIMEDES 305 AND 310

Before you can fit any podules to a 300 series Archimedes, a backplane is required. Our new four-slot backplane may be fitted to any 305 or 310 Archimedes computer, and accommodates up to four single width podules. Installation is very straightforward, and no soldering is needed.

- ★ Accepts up to 4 podules
- ★ Ideal for use with the Computer Concepts RAM / ROM Podule
- ★ Money-back guarantee
- ★ Available as an upgrade

UPGRADES

If you already have a 2-slot backplane, then our 4-slot version may be obtained at a discount price. This offer is for a limited period only - please telephone for details.

Prices (excluding/including VAT)

4-way backplane (incl. fan)	£58.00	£66.70
4-way backplane (no fan)	£48.00	£55.20
2-way backplane (incl. fan)	£33.00	£37.95
Computer Concepts ROM Podule	£46.95	£54.00
As above with battery back-up	£55.65	£64.00
4-way backplane as upgrade	£31.50	£36.22
62256 32K RAM chips	£11.00	£12.65
Postage	£1.00	

HOW TO ORDER: Cheques (payable to "IFEL"), POs or official orders welcome. Access orders accepted on (0752) 847286. Please allow £1.00 P & P.

IFEL, 36 UPLAND DRIVE, DERRIFORD, PLYMOUTH PL6 6BD (0752) 847286

Help!!! needed and offered

• **ARM Code corner** – How about having, amongst our other regular series a section where people could publish useful ARM code routines they had written? It needs someone to receive the contributions and edit them. Any offers? (Start sending in your routines anyway because on past performance I'm sure there will be someone willing to edit it for us.)

• **Sony Multi-sync monitor** – Has anyone had any success using the Sony CPD 1402 multi-sync monitor with Archimedes? (N Edwards, Ferndown)

• **Hewlett-Packard Paint-Jet** – Has anyone got a screen dump for the Hewlett-Packard Paint-Jet colour printer? (N G Edwards, Ferndown)

• **Screen Dump for Star LC10** – Is there one available? R Banks. or a FWP driver? A.M.Guile.

• **Remote sensing** – does anyone fancy helping out with either just general advice or specific help on a remote sensing project – temperatures, water levels, humidity etc, logging the data and also setting up alarms for out-of-range values? (George Foot, Oxted, Surrey.)

• **Pipedream** – we've had a couple of requests for beginners' articles about Pipedream – especially the spreadsheet side of it. Any offers?

• **Fractal programs** – D.P.Allen, Camberley, asks for help with a program that would load a fractal screen from disc and convert the fractal 'contour map' to a 3D perspective landscape (to fill the screen). 'Sea and lake' levels and contour colours should be alterable. He also asks for help in modifying the machine code Mandelbrot program so that the formula can be altered.

• **512 x 512 resolution?** I understand it is possible to get a resolution of 512 by 512 on a normal colour monitor using interlacing. How can it be done? António Pedrinho, Portugal.

• **Sheffield User Group?** – Is there a user group in or around Sheffield? Russell Banks at the Children's Hospital Intensive Care Unit has all sort of questions about using his Archimedes and would value mutual support and encouragement.

Help answers

• **Keyboard Extenders**, as requested last month, are available from Videk Ltd, Unit 10, Bowman Trading Estate, Westmorland Road, Kingsbury, London, NW9 9RN. They cost £7.95 plus £1.50 p&p. Also, Mike Harrison points out that they are the same cables as used on the IBM PS/2 and the Mac SE, so you may do better to ask for one for those machines. (*But I doubt they'd sell them for less than £8.45 inclusive! Ed.*)

• **Elliptical circles on screendumps.** Gerald Fitton's minidump was designed for FX and FX+ printers on which it does give circular circles. However if it is run on LQ's and compatibles (for which it was not designed) you get elliptical circles because the pin spacing is different. However, you can use mode 39 on the LQ printer in which case you get the circular circles back. The alternative is to buy Abacus Training's DumpLQ for just £5.

• **Econet** – To copy the directory structure and files from an ADFS floppy to a fileserver:

```
*COPY ADFS::<drive> NET:<destination directory> ~CFRQ
```

The Q option should be used as it is not only quicker but it avoids a bug in the NetFS when copying chunks of a file. The net directory must NOT exist already as, unlike ADFS, NetFS rejects the creation of an already existing directory. The only way to copy files to an existing directory is to copy each of the contents of the directory in turn.

```
*COPY ADFS:dir1.* NET:dir1.* ~CFRQ
```

```
*COPY ADFS:dir1.dir2.* NET:dir1. dir2.* ~CFRQ
```

To transfer between DOS and Econet, log onto the net first, then do:

```
*ADFS
```

```
PC.emulate
```

```
<insert the dos disk as prompted>  
putfile <dosname> net:<netname>  
getfile net:<netname> <dosname>
```

To save hassle, put copies of getfile and putfile on the disc containing the data. You must not use

PC.PC or a boot file that kills the Econet modules, or does an RMtidy, as this will break the fileserver connection and you won't be able to transfer files! The reduced memory doesn't really matter when transferring files.

- **Solidisk Teletext Adaptor (2.1, page 8)** – Folk said they had tried complaining to Solidisk about their Teletext adaptors, so I wrote to the managing director of Solidisk saying, basically that folk had complained but had no replies to their letters.

At the time of writing, I have not yet heard from them, but I've heard from a reader who has actually managed to get a response from them.

David Howe writes: Solidisk tell me that the reason you cannot now download Telesoftware is a problem with their Archimedes software (I have version 1.20), which is to be re-written, but they don't know when this will be done. However, the software for the BBC (and the Amiga) is said to have been updated and to work OK. Since I was planning to transfer Teletext operations to my Beeb anyway, I purchased the lead and disc for it (£15 inclusive). The TTX program included seems to handle Telesoftware without any problems, although it does refuse to download IBM programs, which the literature suggests it should be able to do. The Archimedes version of TTX used to manage it, anyway. The disc includes the Advanced Teletext System (ATS) program, which can be loaded into sideways RAM, if you have it. I don't, so haven't been able to try it. A ROM version is also supposed to be available, presumably at extra cost. I imagine an Archimedes ATS is bound to appear eventually.

It seems neither of the BBC programs would run on the Arc under the emulator (even given the I/O module to provide the user port required by the adaptor) because they address the port directly.

- **Colour Monitor problems (Archive 2.1, page 8)** – I've also had odd cracks from my Archimedes colour monitor (it seems quite common. Ed.) with an attendant jump in the display. It does seem to be an EHT flashover: as I used to work for a company which manufactured EHT assemblies for colour TVs and monitors I recognise the symptoms. In my case it sounds as though it's within the tube itself, and tends to occur after the machine has been on for

some hours. Without wishing to spread alarm, my dealer's first example had a similar condition which gradually deteriorated and eventually it failed completely (admittedly after pretty intensive use). My monitor sometimes also produces a faint 'baking electrics' smell, which suggests that something is running pretty hot. Is there an incipient weakness in these monitors? Dave Howe.

All 5 Archimedes that I know of, have problems with arcing – usually in damp weather. I had a similar problem with my old Microvitec which, like the Archimedes monitor, stopped doing it when the central heating came on. Apparently the EHT is up near its limit, so any dampness in the air is low resistance to it. Perhaps that's why they named it the ARChimedes. A.N.Other.

Ed: A number of folk, though fewer than those with arcing problems, have written to say that they are getting bad colour alignment at the edges of the screen. This seems to be happening more as the monitors get older, so I suspect this problem will increase in frequency. To see if you've got the problem, do a VDU19;7;0; which turns the background white and then play with the horizontal centring. Any pattern of non-whiteness will be seen quite clearly because as the white area moves, the non-white patch stays in the same place.

The answer, in my view, to both these two problems is to do what most of us haven't bothered to do and that is send them back under warranty. That way Acorn will be able to get back at Phillips who supplied the monitors and tell them to improve their quality control. The trouble is, it's not a catastrophic problem, and no-one likes to be without his/her Archimedes while Acorn finds a replacement.

- **Disc Drive power leads** – Peter Sykes is offering to make up leads so that you can run external 5.25" drives off the Archimedes power supply – £3.50 each. 96 Lanehouse Road, Thornaby, TS17 8EA.
- **Microlink Multispeed modem** – we've so far had three letters explaining how to connect it up to the Archimedes – all of them giving different connections!!! If anyone wants to act as a guinea pig to work out from the three of them, the best way to do it, let us know. If you just want to try linking up your modem, I can send copies of the three letters. **A**



REPTON 3



Around the World in 40 Screens



Repton 3



Around the World
in 40 Screens



The Life of
Repton



Repton Thru Time

ARCHIMATIC REPTON 3

The Thinking Man's Arcade Game

The Repton series of games is the most successful software ever published for the BBC Micro/Electron computers and has only been described as "the thinking man's arcade game". Now, a massive enhanced version is available for the Archimatic, comprising four complete games:

- * Repton 3
- * Around the World in 40 Screens
- * The Life of Repton
- * Repton Thru Time

Enhanced game features include:

- * A choice of four tunes
- * A custom Save/Recall facility

In addition, the software includes a sophisticated **Screen Editor** and a **Character Editor** with a choice of 16 colours from a total of 256. The unique feature is that you can design new screens on the Screen Editor and immediately test them by entering the game section.



The Screen Editor
(designing on Egyptian Repton screen)



The Character Editor
(try the Repton character on different colors)



Acornsoft is a registered trademark of Acorn Computers Ltd. Superior Software Ltd is a registered user.
Dept ARI, Regent House, Skinner Lane, Leeds LS7 1AX. Telephone: (0532) 459453



Please make
or check
payable to
Superior
Software Ltd



24 HOUR TELEPHONE
ANSWERING SERVICE FOR ORDERS

- All major credit cards are dispatched within 24 hours by first-class post.
- Home delivery and packing is free.
- You can pay by cheque and it will be honoured immediately.

Matters Arising...

• **High speed SpellMaster** – we said last month in the Software Available section that Spellmaster and the Inter series are now available on disc – that much is true. Then we said that the disc versions all “still work under the 6502 emulator” and there we were wrong. Archimedes SpellMaster never worked under the emulator – neither the ROM version nor the disc version – both are written in native ARM code. In fact, Computer Concepts have taken up the speed race and are trying to make SpellMaster the fastest spelling checker on the Archimedes. We’ll let you know how they get on!

• **SID upset** – SID’s manager, Philip Colmer, was upset by our comments in the review of the PC Show two month ago. The reason that it was a Master running SID on the stand and not an Archimedes was that it was running an Eonet version. If we had enquired, we would have discovered that an Archimedes nearby was running the real SID, via Fastrak. Philip would like us to point out that comms software is available for the Archimedes and that it works (see the reviews in August & September of Arc-Term U-Connect and Hearsay) and that SID is alive and well unlike certain other Archimedes bulletin boards (ouch!).

As regards our criticisms of the high cost of SID, Philip says that he’s changed the pricing structure. Basically, the £10 a quarter subscription has been done away with but time charges are still 8p per minute. However, there is a minimum quarterly charge of £10. Also, there are now more areas open to guests on 0223-243642. Sorry, Philip, we’re trying to be helpful, honest!

• **Advertiser’s Error** – Computer Assisted Learning Ltd would like to point out that the price quoted in the November issue of Archive for their Archimedes art package, ‘Art Nouveau’ was incorrect. The price should have read £42.50 inclusive. They say that they will, however, honour any orders taken from subscribers to Archive before December 10th, 1988 at the price stated in the advert.

• **What happened to Leonardo?** Leonardo from Beard Technology has changed its name because

another company has put in an application for a trademark on ‘Leonardo’. (Do you think I should try to get a trademark on ‘Paul’?! Ed.) It is now called “The Beard Technology Precision Picture System” – PPS256 for short!

Mike “Beard” Williams tells me that one company making TV documentaries has bought PPS256 (£19.50) because they’ve got an assignment that requires something that their Quantel Paintbox (£100,000) won’t do! They use PPS to do the drawings and then use Paintbox to increase the number of colours and apply anti-aliasing.

• **5.25" interface problems** – Mike Harrison has designed an interface which Watford will be selling in a month or two’s time which allows any combination of one or two internal drives and one or two external drives. An external switch allows logical drives 0 and 1 (the only ones accessible by the PC emulator) to be two internal drives, one internal and one external or both external. ‘Problem’ drives which don’t generate a ‘ready’ signal are also catered for.

• **Mandelbrots on the A440** – William Doggett says the reason his program it doesn’t work properly on the A440 is that he used direct memory addressing for speed which is OK if you get the right memory addresses! To ensure you do get the right addresses, add the following lines at the beginning of the PROCINIT procedure:

```
DIM S% 7 : !S%=149 : S%!4=-1  
SYS"OS_ReadVduVariables", S%, S%  
M%=!S%
```

then instead of the screen address &1FD8000, use the variable M%.

• **Reading non-standard discs** – last month we mentioned the article and program in November Acorn User. John Rickman points out to us that the AU article says that the listing in the magazine is an abbreviated form of the program and that the full version is on their November disc. He typed in the listing then sent off for the disc and found that it was exactly the same program. Grrrr! **A**



Archimedes ART NOUVEAU

The New Art Package
by Barry Christie

ART NOUVEAU

A quality art package specially developed for use by the professional artist and the home enthusiast alike.

ART NOUVEAU is a versatile, easy to use, menu-driven package which offers the user a choice of more than 100 different options including:

Facility to load Mode 12 and Mode 15 screens from other packages.

Bend, Wave, Squiggle and Shear brush options.

Pixel Editor with up to x16 magnification.

Versatile Curve drawing routine.

User defined Screen Grid.

Scratch Pad.

256
COLOURS

ONLY £42.50 inc. VAT and P&P.

Note: ART NOUVEAU requires at least 1 Mbyte of memory.

Order now on our 24-HOUR ACCESS CREDIT CARD HOTLINE on (0698) 733775 or send a cheque/postal order for £42.50 made payable to 'COMPUTER ASSISTED LEARNING LTD.' at the address below.

Please send me ART NOUVEAU package(s) at £42.50 each.

Name:

Address:

Postcode:

Payment is by cheque/P.O. No:

Please debit my Access account card number:

Card expiry date:

Signature:

COMPUTER ASSISTED LEARNING LTD.
Strathclyde Business Centre
Princess Road
New Stevenston ML1 4JB



SEND SAE FOR FURTHER INFORMATION

Readers' Comments

• **AIM defence** – I would like to defend the A.I.M. package reviewed in November. I tried it out and was prepared for the worst because the review said it left you in an unrecognizable environment. However, I found that it did nothing of the sort. It returned to BASIC with no change in the ram configurations at all, even when quitting by <ctrl-break> instead of using the Exit option. However I have found it difficult to load screens from other sources and if any one could help I would be most grateful. (A Bilsby) (*I suspect that Lindis, who distribute A.I.M. have added a re-configuration routine! Ed.*)

• **Monitor Information** – Some people with NEC Multisync II monitor (and maybe other similar multisync monitors) and earlier Archimedes computers may have noticed that there is a green tint to white and grey areas. This is caused by Acorn putting the video sync on the green line. This can be remedied by taking the computer to your dealer and having two FCO or Field Change Orders carried out on it. These are FCO 28 and FCO 29 and are a matter of removing the green sync by taking out several resistors and replacing some by new ones. This change should be free of charge. On later models there should be no problem as the modification is done in manufacturing.

You can also get some distortion of the top few lines in modes 18–20 as commented on last month. NEC admit this problem and say there is nothing that can be done as the Archimedes output is slightly different to the IBM output which is catered for. I suggested the modification quoted last month and they said that the splitting of the sync signals would probably work but achieving the splitting would be the problem. Cutting the legs of chips on the mother board is not to be recommended as it would nullify the warranty and Acorn may have or might come up with an FCO as they did for the green sync to solve the problem.

I would like to compliment NEC's technical staff on there help given to me in solving the above problems especially the green sync, as they got in touch with Acorn directly to get the solution, which several dealers I contacted were not able to do!

(A. Bilsby)

• **Program listings** – In response to your question on the matter of typing listings from magazines I would like to remark that I have always had difficulty doing that no matter what format was used by the editor. The problem was mainly that I had to keep looking back and forward between the listing and the keyboard. One of the programs I found in the latest Acorn catalogue was a typing tutor. I ordered the program from the producers, Contex Computing.

After just one week (spending about half an hour each day practising) I can find most of the keys without looking at the keyboard. This is made possible by Acorn's tactile marking of the 'F' and 'J' keys, a feature sadly lacking on many (even expensive) typewriters and mainframe terminals. If you do get lost on the keyboard, you can regain your 'home position' without looking, i.e. without losing your place in the listing or other original that is being copied.

On startup you can customise the program to react as you wish to typing errors, you can choose to allow varying percentages of typing errors, you have the option of whether or not you can use the delete key to correct mistakes, and you can vary the typing speed required to graduate to higher lessons. With the default settings of <8% errors sufficient to progress to the next lesson and with the delete key enabled I got to lesson 47 of the 90 in just three days. I then changed the settings to 0 errors allowed, and no use of the delete key possible. This I can only recommend to perfectionists or masochists!, if you rest your fingers too heavily on the keys while searching for the home position, you will have made an error and will have to repeat that lesson.

The single disc with just 38k of information comes in a sturdy holder very similar to those used by A&B computing for their 'red keys' discs. There is just one sheet of paper with information on how to configure your Archimedes for the software and all other instructions are on screen. These instructions are clear and concise. The program is easy to use and effective. I have used computers for years both at home and at work without learning to touch-type. At last I have hopes of being able to learn relatively

easily, and (not counting the purchase of the Archimedes) for a very reasonable price. The program costs £15.00. I don't know whether postage is included for the UK; I was charged £1.50 for postage to Holland. The disc is not copy protected; Contex do warn though that each copy is secretly coded so that illegal copies can be traced to their original buyer who "will be liable to recompense the software supplier for lost income for all such 'pirated' copies discovered". (P Green, Holland)

- In defence of ROM/RAM podules** – There has been some correspondence recently concerning the use of the ROM/ROM board in the Archimedes. It seems some owners are not sure what advantages these boards bring, even to the extent that one letter said "it's all a con by Computer Concepts" or words to that effect.

Needless to say I do not agree with this opinion – on the contrary I believe these boards have some very real advantages to Archimedes owners. For example;

- Convenience.** One of the large benefits that ROM software has always had, on the Archimedes and the Beeb, is that programs can be permanently available. On my machine I can type *IWORD and be in the word processor instantly, I don't have to be in the correct directory, or find the right floppy etc. OK, the same benefits are possible from a hard disc, but at a considerably higher cost.

- Space.** Consider the situation of a suite of integrated applications – say a word processor, graphics, spreadsheet, spelling checker and thesaurus. In no way will it be possible to hold all of these in RAM and so run them together, even on a 1M Archimedes. It is our intention to have all these running from ROM on an Archimedes by using program overlays and accessing the data in ROM. While this is in the future, currently available ROM programs such as SpellMaster already benefit by using this technique. The ROM based versions use considerably less RAM than disc based versions.

- As a filing system.** The Computer Concepts ROM/RAM podules are provided with a very fast ADFS-like filing system. These boards are not just ROM boards like the ones available for the Beeb, but have the ability to have RAM chips plugged into any socket providing a RAM filing system. This is

not the same as the RAM filing system that will one day be available with RISCOS since a) it can be battery backed and b) it does not use main RAM, so leaving this scarce resource available for more important things, such as application workspace.

This filing system, while not having the capacity of a hard disk, is very fast. Here are some example timings (in milliseconds)

	Hard disk	Floppy	RFS
*SAVE 20k	41	181	24
*LOAD 20k	22	140	9
*SCREENSAVE 80k	1002	2530	92
*SCREENLOAD 80k	574	1830	65

As you can see it is considerably faster than even a hard disk, and TEN times faster than floppy discs. A practical example of the benefits possible is the linking of twenty or so object files that make up large programs, such as our Archimedes word processor. This is at least twice as fast on the RFS as it is with the hard disk and so saves us considerable amounts of time in the development cycle of a product. It goes without saying that we all use the RAM filing system in our development work wherever possible.

I believe there is a place for ROM/RAM podules in the Archimedes. It has some unique advantages of its own and at the same time offers many of the benefits of a hard disk, especially in filing system intensive work.

While we have now released our products on disc, I do still feel the ROM versions are more attractive and I am sure this will always be the case, probably more so, with future releases. (Charles Moir, Computer Concepts.)

- GammaPlot by Minerva** – This program has much to commend it and meets many of the claims of Minerva except for two quotes on the back of the box. It is not, in my view, suitable for 'business graphics' nor is it 'easy to use'.

I have used Easel on the QL for a number of years to very good effect in preparing 35mm visuals to support business lectures. It came free with the QL and is available as part of PCFour (on IBM-PC compatibles) which costs about the same as Gammaplot if you shop around. Mind you, the

Readers' Comments

graphics on the PC, or under the emulator on the Archimedes, are not as good as the QL and certainly do not match the potential of the Archimedes in its native mode.

GammaPlot was to replace Easel and capitalise on the Arc's superior performance to the QL. I was to be disappointed because the machine's advantage was overshadowed by the inferior software. These were my main findings;

1. No support for MODE 20.
2. No horizontal label option on line graphs with vertical figures.
3. Only one graph per chart. Overlaying is a pain to arrange.
4. The same colours are used for each graph.
5. No automatic Key or Titling (for the graph) is provided.
6. Data entry is tiring as the cursor is never in the right place.
7. No true three dimensional graph formats. i.e. x,y and z axis.
8. No vertical labeling or rotational ability.
9. No install program to configure drives. This means that each time you save a file to a different drive the program loses track of its overlays.
10. Unlike Easel and most PC software, GammaPlot is copy protected which adds to the aggravation in use and may account for the messy implementation. (Incidentally the modeling program Euclid which is largely written in BASIC V is not only a much tidier program but it is a good deal faster, and works in Mode 20). The speed of disc access on Gammaplot is often downright pedestrian.

In conclusion I must admit that in my evaluation of the program I may have got some things wrong. The manual is well written but the program is excessively complex, however the macro facility looks excellent. It's also true that if you take enough time over it, it is possible to achieve excellent pictures, but this does not change the fact that this is not a professional graphics package. The authors would do well to look at the competition on other PC's. The bells and whistles seem to have distracted the authors from providing the minimum requirement in generating basic graphs. (Brian Oliver, Kent) A

Hints...

• **Fontsize problems?** (Ed. got caught out on this one) With Clares' Sound and Graphics demo, when you start it up, it says it needs 44k of fontsize, so I tried to re-configure with fontsize 6 on the basis that 6 times 8k = 48k. Now, as I'm sure you will all realise, that didn't work because fontsize is specified in 4k chunks, not 8k or 32k.

• **Logotron printer dump** – Logotron told one of our readers that there is no printer dump for their Archimedes Logo. However, our Printkey Dump program (issue 1.10, page 45) works fine.

• **RX80 dumps for Artisan** – Clares originally said that Artisan wouldn't dump pictures on an Epson RX-80 and that the FX-80 was the 'baseline' printer as far as they were concerned. However, they have since produced a 'help-sheet' of the pokes needed to convert the dump to an RX-80.

The gist of the change is as follows;

If your printer will support CRT mode graphics, (ESC "*4) then use it. If not then use ESC "K" for normal single density bit image mode, but CRT mode gives a much better image and less distortion.

```
*LOAD ART6 10000  
!&10EC5=&042A1B for CRT or =&4B1B00 for  
single density  
*SAVE ART6 10000 +12FF  
*SETTYPE ART6 &FFB
```

The resultant print produces oval circles(!) but can be triggered from Artisan.

• **Monochrome displays on the 440** – Brian Cowan writes, I was lucky enough to have one of the first release of 440's which I used with one of the Acorn colour monitors. I know that the resolution of these monitors is nothing fantastic, but it seemed sensible to purchase them together; at that stage I was not sure what connection standards were used.

About six months later I had reason to use an ordinary mono monitor on the 440, and imagine my horror when I found it did not work. Ordinarily I would have sent the computer back to the supplier to have it fixed under warranty. However, the machine was in constant use so we decided to live with it, using only a colour monitor.

and Tips

I now discover that there is nothing wrong with the 440!! When these machines leave the factory they are not configured to drive standard resolution mono monitors, they are set up for high res monitors. Inside the case, on the PCB are some jumpers that must be set according to what sort of monochrome monitor you want to drive.

At the north-west corner of the board is a plug called PL2. I think this does nothing except carry some pin connector links. A little further southwards on the board are some pins labeled LK13 and LK14. If you want to drive a standard resolution monochrome monitor, you must take two links from PL2 and place one on LK13 and the other on LK14.

Connection to a monochrome monitor on 440 machines is through two BNC sockets on the back panel. One socket is marked SYNC and the other is labeled MONO. For high resolution, both of these are used but for standard resolution only the SYNC socket is used. Since this is a BNC socket rather than the phono socket on the 300 series, you will have to make up or purchase a special lead or a converter.

For those of you contemplating the purchase of a multisync monitor for those extra modes etc. you will be happy to hear that the Archimedes connections are the same as those on PC machines. This means that if the monitor comes with a lead, it will probably be suitable for the Archimedes. This is certainly true for the Taxans that I use. Incidentally, concerning the Taxans, some have rather dull pictures. Has anyone tried brightening them up?

- **Desktop calculator** – “The calculator on my desktop doesn’t respond to the keyboard whereas your review of Arthur 1.2 said it did.” So I tried to work the calculator with the keyboard keys myself and it didn’t seem to work. Then I realised that it won’t respond until you enable it for keyboard input by clicking on it with the mouse. The top bar then goes red and it’s ready to take input from either the main keyboard or the numeric pad, delete being the equivalent of clear. Easy when you know how.

- **Conditional booting.** If you have a boot file in the RFS on battery-backed RAM, you may want to do some things when you first switch on but not

every time you do <ctrl-break>. You can achieve this by making it conditional on the monotonic timer, i.e. the timer which is initialised at switch-on and not reset in any other way. To read this timer, use SYS &42 TO T% where T% then gives the time in centiseconds so you can say that IF T%<200 (say) then do the switch-on bits and pieces ELSE do the <ctrl-break> things.

- **CharDes fonts** – If you have fonts produced by CharDes, you can convert them for use on the Archimedes as follows. Enter 65Arthur and type

```
*spool thin2
*thin
*spool
```

where thin is the name of the original font file and thin2 is going to be the new file. Then use a text editor (such as the Master128’s “Edit”) and edit out the first and last line (i.e. the *thin and *spool). Then resave as thin2 and settype it as &FF7 (BBC font). When you need it, just type *thin2. If you do a *SHOW, you will see that the load and run actions of a file type &FF7 is to *PRINT it – which is just what you want.

- **Command files** – In order to disable vdu output during a command file e.g. the !boot file, try:

```
*echo ||U
...
*
*echo ||F
```

Two string escape characters are needed (as I’ve explained in a previous tip) so that the final command executed is *echo !U

i.e. <ctrl-U>. This issues a VDU 21 which disables vdu output until a VDU 6 is issued <ctrl-F>.

This is similar to the ECHO OFF command of MS-DOS. Indeed if the string used is:

```
*echo ||U|H|Hoff
```

then all that is visible is “echo off” which explains why nothing else is visible to the user and also hides the control code sequence used. The “compiled” string is then *echo |U<8><8>off. One side effect is that two line feeds will occur so if this is undesirable a couple of reverse line feeds can be incorporated with

```
*echo |U|H|Hoff|K|K
```

Hints and Tips

Note that the use of single string escape characters means that the cursor control codes will have immediate effect whereas the <ctrl-U> will only be issued when the echo command is executed.

The double string escape characters are only needed for use with *Build, if a text editor is used then only single characters are needed but the control codes must be input directly, e.g. so they appear as inverse video in TWIN, for those cases where the affect is immediate.

(Sorry if this sounds a bit confusing but the difference between two and one l's, and when to use which, is an entire discussion point on its own. Perhaps this could be a short article – it is important whenever command files issue messages etc.) Clifford Hoggarth.

- Quick *COPYing and *WIPEing – When you enter a wild card *COPY or *WIPE on a list of files such as TEST1, TEST2, TEST3, etc. you are presented with a list of options at the end of the copy or wipe statement

```
*COPY :0.TEST* :1.TEST*
Copy file adfs:::0.TEST1 as adfs:::1.
TEST1 (Y/N/Q/A)?
etc.
```

The option Y, N, Q and A mean the following :-

Y – means copy that file

N – means don't copy that file

Q – means quick copy all file with the wild card specification after and including that one with no further confirmation prompts

A – means abort from copying

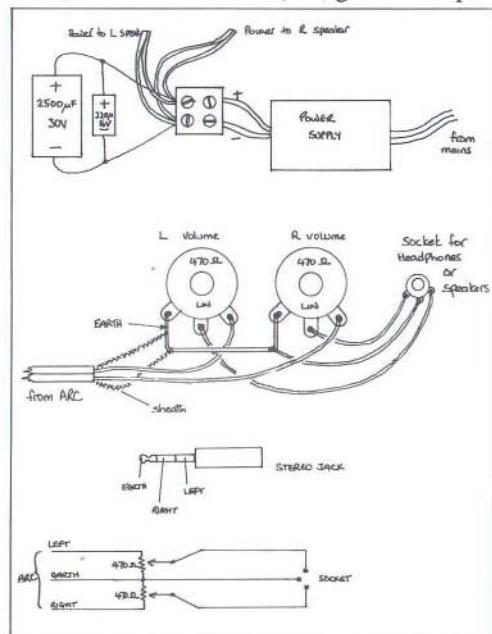
The *WIPE option works in a similar manner but deleting.

- Stereo Speakers – After reading the hint in Archive 1.2 about adding a pair of Tandy amplified speakers, I promptly rushed off 'to town'. They had just run out of stock, because the £25 price-tag was a special offer. However Tandy were very helpful and ordered me a pair which arrived within a week and which at £29.95 are still excellent value. However, a set of alkaline batteries was quoted as £7.95!! Instead, a surplus old calculator mains adaptor and two power plugs (30p each from a local electronics shop) provided a cheaper source of

power. If you don't have a spare one, a new power supply from Rapid Electronics costs less than £4. The magic moment arrived and I switched on! The buzz from the speakers was horrific!, but soon drowned by mayhem and destruction from Zarch, which in turn was swamped by the dulcet tones of 'her ladyship' complaining from the other end of the house about the volume!

Back to the drawing board – two capacitors, recovered from an old tape recorder smoothed the power supply output and banished the buzz. The solution to the lack of volume control was to add two potentiometers. Whilst I was at it, I decided to wire the volume controls into a socket, so that Zarch could be enjoyed at a reasonable volume late at night through a pair of personal stereo headphones as well as through speakers at a more civilised hour. An hour's work, a couple of pounds and I now have glorious stereo sound, at a reasonable volume!

The diagrams below show the set up. A twin pot could be used instead of two single ones, so that both speakers are controlled from one knob. Any value pot from 100 ohms up to about 4.7k ohms will work, but 100 ohms is ideal, though 470 ohm pots



are easier to obtain. Logarithmic pots are best, but linear ones work fine and are again easier to obtain. The big capacitor can be anywhere in the region shown (2500 microfarads/30V) – the small one takes out the higher frequencies and again its value (220 microfarads/16V) is not critical.

It's worth noting that the speakers cut off only after a minute or two if there is no sound being produced. Presumably there is a small capacitor inside which needs to run down first. Similarly, they need a certain minimum signal for a very short time in order for them to switch back on. David Kent

- **Ultra-cheap sound amplification** – Maplin Electronics (0702-554161) have an amazing offer including a pair of walkman-type headphones AND a pair of monitor speakers for just £3.95! I don't know how good they are, but at that price, you can't go far wrong. (Oh, there's a handling charge of 50p and a postage charge of 50p – still, it's not bad.)
- **View "OC" Command** – In View B3.0, one of the highlights can be set to 27 instead of the normal 128 for underline and 129 for emphasise. In doing this, commands may be made directly to the printer as in the Wordwise OC command. For example, to set highlight 1 to 27

<Shift-f8> HT<return> 1 27 <return>

On pressing <f4> (underline) the next character will be sent to the printer as a command, i.e. ShFn4x1(-x1) will turn on NLQ mode (on some Epson compatible printers). This system seems to work for most commands, but some commands need an ASCII ,1 so to get round this set highlight key 2 (emphasise) as 1 and type ShFn4wShFn5 (-x*) to turn on double height for example. This does have the disadvantage that things can't be turned on and off in the same line as you have to set highlight 2 back to 0 to turn it off.

- **Two ARM Code Assembly Macros** – (Richard Averill) – Here are two useful macros for use in the BASIC V ARM assembler. They both use the same method of loading 32-bit words into a specified register, which is detailed below:

Originally, I had the idea that if I used the following piece of code:

```
LDR <reg>, [PC, #-4] !
EQUD <value>
```

then the processor would load the value and jump on to the next instruction. However, what actually happens is that the value is loaded and the ARM tries to execute the value as an instruction. If the high-byte of the value is zero, then this will not cause a problem, but it is not advisable to use routines that are not totally water-tight.

Here is the improved code:

```
LDR <reg>, [PC]
MOV PC, PC
EQUD <value>
```

which only takes one more word of code and works all the time. The MOV PC, PC is there to make sure that the ARM executes the instruction that is in the pipeline, and not the instruction in <value>.

Using these ideas, I have developed two useful routines detailed below:

```
DEF FNload(reg%, val%)
[ OPT opt% AND &E
LDR reg%, [PC]
MOV PC, PC
EQUD val%
] : =opt%
DEF FNadr(reg%, adr%)
[ OPT opt% AND &E
FNload(reg%, adr%-P%-20)
ADD reg%, PC, reg%
] : =opt%
```

As you can see, the macros can be used as any other instruction in the assembler as such:

```
MOV R0, #123
FNload(1, &12345678)
FNadr(2, pointer)
SWI "OSAnything"
```

A demonstration program is given (on the program disc) to illustrate the use of these macros.

- **Running ViewPlot under 65Arthur** – Richard Averill – Listed here are the changes needed to make ViewPlot 'harness the power of the Archimedes!'. They will tidy the programs up, allowing ViewPlot to run in any mode and to print and save screens. A *Exec file of these changes is provided on the monthly disc.

To use this file, transfer the ViewPlot disc onto ADFS into a directory such as 'ViewPlot'. You can either create a text file of these commands (with

Hints and Tips

*Build or a text editor) or you can type the commands in yourself. Either way, you should make sure that you are in the ViewPlot directory when you try to convert the programs.

*|Running ViewPlot under 65Arthur.
(C) Richard Averill, 1988.

*BASIC
LOAD "ViewPlt"
390
SAVE "ViewPlt"
LOAD "V_D"
730 IF M=248 OSCLI ("Screensave Image")
SAVE "V_D"
LOAD "V_M"
1140 DEF PROC0:VDU 28,0,23,39,5:CLS
1190 REPEAT
1200 INPUT "Enter screen mode (not
text) ? " M%
1210 UNTIL FNcheckmode (M%) =TRUE
DELETE 1220,1290
3000 DEF FNcheckmode (mode%)
3010 LOCAL col\$,ok\$
3020 ok% =FALSE
3030 IF mode% =0 OR mode% =8 OR mode% =12
OR mode% =15 OR mode% =18 OR mode% =19
OR mode% =20 THEN C% =16:ok% =TRUE
3040 IF mode% =1 OR mode% =4 OR mode% =9
OR mode% =13 THEN C% =32:ok% =TRUE
3050 IF mode% =2 OR mode% =5 OR mode% =10
THEN C% =64:ok% =TRUE
3060
3070 col\$ = "04"
3080
3090 IF mode% =0 OR mode% =4 OR mode% =18
THEN col\$ = "04"
3100 IF mode% =1 OR mode% =5 OR mode% =8
OR mode% =19 THEN col\$ = "15"
3110 IF mode% =2 OR mode% =9 OR mode%
=12 OR mode% =20 OR mode% =10 OR
mode% =13 OR mode% =15 THEN col\$ = "2"
3120 IF ok% =TRUE THEN OSCLI ("LOAD Col"
+col\$+" 2600")
3130 =ok\$
SAVE "V_M"
LOAD "V_P"
250DATA 0,0,0,0,0,0,2,4,4,4,2,16,8,1
,16,0,0,0,2,2,16,4,1,16,0,0,0,2,
1,16,0,0,0,0,0,2,8,2,2,4,4,2,
2,16,2,1,16,2,8,2,2,8,2
1740DEFPROCJ (b%):e% (b%) =0:ONM% +1GOTO
1770,1810,1880,1880,1770,1810,1880,
1880,1810,1880,1880,1880,1880,1880

,1880,1880,1880,1880,1770
,1810,1880,1880,1770,1770
2070DEFFPROCE:FORb% =1TO8:e% (b%) =b%?
(&2608+((a%-1)*2+a%-10)*8-1):ONM% +1
GOTO 2110,2150,2200,2200,2110,2150,
2200,2200,2150,2200,2200,2200,2200,
2200,2200,2200,2200,2200,2200,2200,
2110,2150,2200,2200,2150,2150
SAVE "V_P"

- Cheap colour monitors – Peter Sykes bought a Commodore 1084 monitor which seems to be identical to the Phillips 8833 but cheaper. He got this from Eazypoint in Staines, Middlesex for £240, around £20 less than the cheapest he could find the Phillips 8833. (There's an Acorn one for sale in the small ad's, but there must be other folk by now who are up-grading to multi-syncs and have monitors for sale – send in your small ad's folks – no charge.)

- Use of Archimedes on Econet – Copying ADFS format discs to NET and vice-versa, no utility is needed as Archimedes has it all built in:-

COPY -adfs-\$. -net-\$.whatever.* QRC

Here ADFS is the source, NET is the destination. The 'whatever' is the chosen destination directory, or directory path (e.g. replace 'whatever' with 'fred.mary.util') The * is to indicate all files. The Q indicates QUICK copying using all available RAM. This damages resident programs but considerably speeds the copying process. It is also almost essential if using early versions of the Econet Software Modules (NETFS, ECONET) due to bugs in these which cause 'Not Listening' messages. Later versions do not suffer this. The R to indicates recursive copying of sub-directories. The C turns OFF confirmation so that copying occurs for all files without question.

Simple modifications to the command line allow copying in the reverse direction.

(This was sent in by Michael Ryan, editor of the Econet User Group Magazine (NEUS) which carries regular articles on Archimedes on the network (and many less esoteric matters as well). We have also been licensed by Acorn to provide free upgrades to Econet software modules for those unable to obtain them elsewhere. Econet User Group, Balkeerie Cottage, Eassie by Forfar, Angus, DD8 1SR.)

FWP Hints & Tips

compiled by Mike Hobart

- **Changing default drive** – those of you with dual drives may want to be able to get FWP to default to looking for data on drive 1. To do this, all you do is load \$library.1wp and change the line that sets up the documents directory to:

```
*set FirstWordPlus$Docs :1.$.
```

- **Saving to a fresh disk** – If you need to save your work onto a fresh disk, you may have difficulty in persuading the save to work: you get silly messages about the disk being write-protected and directories not being present. This is very frustrating if you have just written the first chapter of your bestseller. Try "Save as...", taking special care either to see that you are in the correct directory as specified in the directory box at the top of the save window or create a new file in the root directory by clicking repeatedly on the X icon until you are left with only a "*". Thanks to Peter Tettmar, on whose solution is different, but less convenient as you have to anticipate the problem!

- **Form feeds in the wrong places** – Edit your printer driver (the "hex" directory one) to deactivate the "vertical tab to line" feature. This is done by placing a "*" at the beginning of the line (line no 4, which should now read *4,1B,42,80,0,B or near). If you have serious troubles with irrational formfeeds, try setting the CONFIGURE IGNORE of the battery-backed RAM to prevent the computer ever sending to crucial code to the printer. (Thanks to David Adamiak and from him to GST Holdings).

- **Use of IBM fonts** – Steve Jones points out that there can be advantages in using the IBM font set, if your printer supports it. The monthly disk contains his very full instructions and programs. The main reason for using the IBM fonts is the availability of certain symbols for maths and graphical characters. These can be especially useful for designing forms.

- **Graphics problems** (e.g. that white is printed grey etc.) can be cured by preparing the graphics in 1st Word palette. For Artisan, copy 1stword's Resources.1wp.!palette as artisan.1wppal (or what takes your fancy) on the Artisan disk, then boot up Artisan, grab the disk ikon and finally select the new

palette. The result is not artistic, but a quick roller brush over the background with white, followed by the artwork in e.g. black produces a picture you can import and have printed correctly. The reason this works is that the graphics dump routine in 1st Word assumes the use of the program's own palette, or at least that it is trying to produce a paper version of what you see on the screen. It is, incidentally, quite an intelligent and quick dump, which checks to see if there are pixels to print before printing a line, and issuing a linefeed if not.

- **Conversion from WWPlus** – The following hint might be useful to people still trying to convert WW+ file to 1WP. The convert program published in Archive does a wonderful job. However, I found it a bit irksome having to edit out all the 'US's and 'UE's etc which are left after the program has dealt with the green and white embedded commands. (This just goes to show how quickly we start to take things for granted!!). I also found that it was impossible to reformat the text to a new line length. The latter problem seemed to be that 1WP ended lines with &OE, whilst converted WW+ files ended with the last letter of the last word on the line.

The solution to both problems is as follows:

- 1 SPOOL out the WW+ file, using Option 8 (This removes all embedded commands and centres text and produces indents and TABs)
- 2 Reload the spooled file and use Search & Replace to change all double returns to @@ (or any other unused combination of letters)
- 3 Change all of the remaining single returns to <space>-return
- 4 Change all of the @@s (or whatever) to <space>-return-return
- 5 SAVE the modified file using Option 1
- 6 Run the SAVED file through the convert program

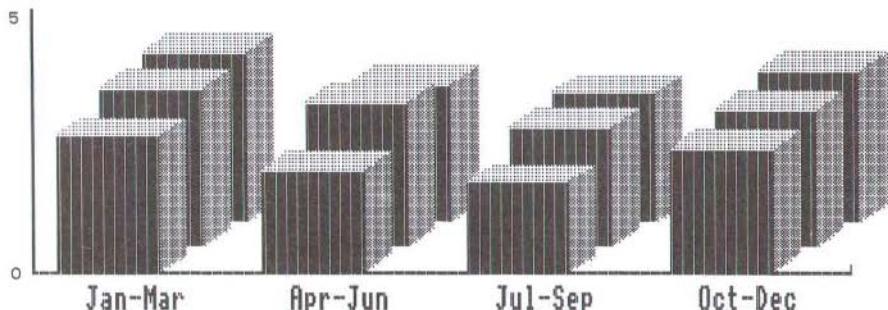
The effect of 2, 3 and 4 is to save the paragraph breaks, but end all of the lines with a space. The result is a file with no extra 'remains' of embedded commands, and which will reformat correctly in First Word Plus. A

Help!! Has anyone got a driver for a NEC P6 or P7 Plus? Geoff Thorington.

PRESENTER

Affordable Presentation Graphics for the Archimedes

PRESENTER is a truly professional graphics program which allows the user to create, modify, print out or photograph high quality colour displays of data in either bar, pie or line format. Data is entered either manually in spreadsheet fashion or can be imported from other spreadsheets or programs in comma separated format. **PRESENTER** makes full use of the Archimedes' WIMP environment and high resolution graphic modes and incorporates 3D displays, auto-scaling axis, user definable layouts and multi-layer graphs. Screens may be saved for use with graphic packages like **ARTISAN** or graphic wordprocessors like **1ST WORD PLUS & GRAPHIC WRITER**. **PRESENTER** can also be used with **PIPEDREAM**.



At only £24.95 ex-VAT. **PRESENTER** is an invaluable extension to existing spreadsheets and wordprocessors. It is a must for all Archimedes' users.

PRESENTER is available from all good Acorn dealers or direct from LINGENUITY.

Please send me... PRESENTER Packs	Retail Price	£24.95	£.....
	VAT	£ 3.74	£.....
	Postage UK	£ 1.15	£.....
	Postage Overseas	£ 6.00	£.....
		TOTAL	£.....
Payment Visa... Access... Cheque...			
Credit card number		
Card Expiry Date		
Name(as on credit card)		
Address		
 Postcode.....		
Send to LINGENUITY, Dept.JT, P.O.Box 10, Halesworth, Suffolk IP19 0DX.		

Telephone orders accepted - ring 098 685 476

What price Pro-Artisan?

Malcolm Banthorpe

Over the past year, Artisan has established itself as probably the most popular graphics/painting program for the Archimedes. It made its appearance at a time when RISC software was generally thin on the ground and its exemplary use of the wimp environment set a standard which some of the other software houses would do well to study. Other painting programs have come (and some gone) in the meantime, some offering facilities lacking in Artisan but, in my opinion, it has up to now remained the best general purpose graphics package. Now ProArtisan has arrived and looks set to become the new standard by which painting programs are judged. To consider it as simply a 256 colour version of Artisan but would be to do it an injustice. A few minutes use of the program proves this and possibly helps to explain why it has taken so long to develop.

Its overall look and ease of use are very similar to those of its predecessor and anyone who has used Artisan will immediately feel at home with it. Like Artisan, the HELP option selectable from the main menu gives a running commentary on what you are or should be doing, at the base of the screen. This means that even in the earlier stages of the program's use, the manual need only be referred to occasionally for the finer points and attention can be kept on the screen. Since most readers will at least have seen Artisan in operation (if not, refer to Archive 1.5 for a review), I will confine this review mainly to those features which are either additional to or different from Artisan, referring to each of the main menus in turn.

The Colour Palette

The first noticeable difference is the way in which the colour palette is handled. It shows 256 colours to be available, all of which can appear on the screen at the same time. This contrasts with Artisan which has 16 redefinable colours available at any one time, selectable from a total palette of 4096. With so many colours appearing on the menu, considerable thought has been applied to the way that they are grouped on the palette display. The result is that

deciding exactly which shade is required for a particular purpose is made much simpler than if they had been merely been arranged in order of their GCOL and TINT numbers. For instance, if you require a range of colours from light to dark blue, it is easily found on the menu. Under most conditions the colour palette window pops up whenever the mouse's menu button is pressed, a second press being required to bring up the current draw/paint/sprite etc. menu. This is sensible in view of the fact that the colour menu is probably the one most frequently required for access. It takes a little getting used to after Artisan but it soon becomes second nature to double-click if you want to go straight to one of the other menus. Colours may also be picked up directly from the picture on the screen.

The Draw Menu

As in Artisan, the Draw menu gives access to brushes of various shapes and sizes. In addition to painting in the currently selected colour, the brush can be made to cycle through the colours in the currently selected 'colour set'. A colour set is a group of any 16 colours and may be selected from a number of pre-defined sets or may be user-defined. The fill function has its own sub-menu offering a number of fill options in addition to the standard flood fill with a solid colour. Linear and logarithmic, stepped and graduated fills in vertical, horizontal and circular formats are available. Graduated and stepped fills again make use of colour sets and the Fill sub-menu allows for their selection and/or definition. The possibilities opened up by these new fills are considerable, making it relatively easy, amongst other things, to render curved surfaces with smooth shading. It is also possible to fill using the currently selected sprite.

There is now a full Spray function with selectable spray size and density. If, in addition to the spray colour, a key colour is selected then the spray will be confined to only areas of that colour otherwise it sprays on any part of the screen. You can also get the spray to cycle through the current colour set.

The slightly oddly-named Wash function, accessed via a 'tap' icon, works by averaging adjacent pixels

and may be used in a number of ways. Applied with a variable-sized rectangular brush, it will get rid of hard or jagged edges by blurring colour boundaries and can also give soft focus effects. The more it is applied to a given area, the greater the degree of blurring. As an example of its use, I found that by spraying white onto a deep blue background and then applying Wash it was possible to produce a nice soft cloud effect.

The Magic Brush function is an enhanced version and allows up to 25 colours to be changed to 25 other colours in a single pass of the brush.

The Banding Menu

There is now a variable grid-lock facility, useful in drawing diagrams and in generally lining up different parts of a picture. The other major addition to this menu is the Bezier curve function, which has its own sub-menu. A Bezier curve starts as a straight line with four control points. The line can be distorted into a curve by dragging the control points. You will probably have come across a similar application in the Font Designer on the Welcome disc. Essentially it provides a much easier way of generating smooth curves than trying to draw them freehand. A Bezier object will consist of one or more joined curves. The Bezier object definitions may also be saved to disc, thus making them repeatable. A previously defined object may also be stretched in a way similar to the rubber pen option in Artisan.

The Sprite Menu

Sprites may now be picked off a zoomed screen window, as well as the normal screen, allowing them to be picked with greater precision.

The Select Sprite icon now brings up a window showing all the current sprites rather than just a list of names.

Irregularly shaped sprites may cut from the screen by outlining them with a rubber-banded line of the selected mask colour. This provides a much quicker alternative to painting out all the background in the mask colour when, for instance, cutting out an object from a digitised image. It also has the advantage that it works non-destructively in that the outline disappears after the sprite has been grabbed. The entire screen contents are left intact.

Sprites may be scaled vertically and horizontally to any proportion. The mask may also be separately scaled and displayed in the currently selected colour to give drop-shadow effects.

The Cut and Paste menu

Facilities are exactly as for Artisan. With the exception of rotation, the same effects can now often be achieved more conveniently using the sprite menu. The one advantage of using the Cut and Paste menu is that no sprite memory is required.

The Tool Menu

There are a large number of monochrome and colour printer dumps available. In addition, alternative dumps may be loaded by calling up an OSCLI window. This could also be used to activate user-defined modules. With this in mind, the facility to define a rectangular screen area, whose coordinates are then available to the module, has been provided.

Distort now allows the current sprite to be distorted to fill almost any user defined shape.

The Global Toner applies the current colour set to a selected rectangle and can tint or 'solarize' the picture depending on the colour set. One of the pre-defined colour sets, a series of yellows, gives a sepia photograph effect.

Global Magic Brush applies the Magic Brush effect to a rectangular area and, as well as being quicker in some circumstances, it allows some effects which are not possible when using the brush.

A simple font editing window may be called up to modify existing fonts or to create new ones.

The Disc Filing Menu

Pictures can be stored either in standard or compressed form. In addition to sprites, fonts and pattern fills, Bezier curves, colour sets and Magic Brush sets can also be saved.

Only a short provisional manual was available with the review copy but no difficulties were experienced in using it. Like its predecessor, ProArtisan is highly intuitive in use and the Help option gives on-screen directions if required.

In reviewing Artisan last February, I stated that it was probably just a taste of the quality RISC

software we could hope to see in the future. ProArtisan certainly lives up to this prediction by containing, as well as all the expected features of a painting program, a number of innovations. The cut and paste facilities and the graduated fills I found to be particularly impressive. Bear in mind that I've described only the new features above and not those features already seen in Artisan such as shape drawing, pixel editing etc.

But now for the bad news: the recommended price is £169.95. No, that's not a misprint; there really is a one hundred in front of the £69.95 - over four times the price of Artisan. So is it worth the money? The answer is yes; the facilities offered really do open up a lot of new opportunities for the production of high quality screen graphics. But the question of whether it's worth it to you will depend ultimately on how much you intend to use it and to what purpose. I think it's possible that Archimedes users have up to now to some extent been spoilt by the availability of some very high quality software at prices envied by users of other micros (compare the price of a similar application for a Macintosh II). Artisan is arguably underpriced and certainly appears so if compared with some other software offerings. All the same, at this price, the market for ProArtisan is bound to be somewhat limited. If you think you may be willing to splash out on the program, or even if you're just interested in graphics, then I strongly recommend that you try to get a look at ProArtisan in action. Just in case you are still undecided, Clare's are offering, as an extra inducement, a limited edition polished wooden artists' palette box with orders placed before release. If, despite this amazing offer, the price seems too high then the original Artisan can still be thoroughly recommended. You could try dropping a few hints about Christmas presents! A

Quazer from Impact

A Bilsby

This is another shoot-em-up game and is the first games release for the Archimedes from Impact. It also works out the cheapest so far at £9.50 to members of their Games Club (£11.95 to others - £11 through Archive). It is supplied on a protected disc but not the usual type that prohibits backing up or copying by corrupting the free space map. The disc can quite easily be backed up, but booting the backup copy load perfectly until the last moment when it stops and requests the master disc to be placed in the drive. The game then starts. Quazer should work on a 305 with no trouble but it does require a screen size of 20. It takes just over 10 seconds to load whereupon you are presented with a neat title page and some very good stereo backing music. You have the option to press space to start, press 'J' jump a level or 'P' to enter a password to a higher level. Leaving the computer alone for a while prompts a vertically scrolling screen giving the background to the game as well as credits.

Pressing <space> reveals a message screen showing what level it is, the password for that level and the number of targets you must hit. The targets are red diamonds in white boxes and failure to hit the required number by the time you reach the end of the level means you lose a life and start that level again.

The playing area fills the screen and is a view from above with the score and number of targets hit tucked neatly at the bottom left and right of the screen, however there is no indication of the number of lives you have left. Your ship is at the bottom of the screen and can be moved by <Z> and <X> for left and right, <Y> and <C> for up and down and <return> fires the torpedos. The action is fast and the vertical parallax scrolling is very smooth in the 256 colour mode. There are a whole host of aliens from simple green spheres and mouse pointers to complex and well detailed sprites which all move very well. Most of the aliens can be destroyed with one shot, others need several and some can not be destroyed at all.

The landscape is also well detailed and has to be watched - there are terrestrial obstructions that you

Contact Box

- Auckland (N.Z.) BBC/Archimedes Users** - There is a club for such as yourselves. Contact Theo Olifiers, ABUG, P.O.Box 9322, New Market, Auckland, New Zealand. (Also on Bulletin board on 5348414, data: 8n1, 300/300 or 1200/75.)
- Civil Engineers**, contact António José Mendes Pedrinho, Rua A à Avenida Melo Falcão, Lote 305E 10 Esq, Pontinha 1675 Lisboa, Portugal. A

must avoid or your ship will be destroyed. One complaint about this is that in some places if you are killed by an obstacle, you are put back just in front of that obstacle on the next life and often so close you can't avoid it a second time. This happens especially in the mine field at the end of level 5. On completing the screen you are given the password to the next level and the number of targets to hit. The screens become more difficult as you progress.

Pressing <J> from the menu allows you to jump to the last completed level, however you start it with the number of lives you started that level with previously so you often jump to a level with only one life left. I do not like this at all but you can get around it by pressing <P> from the menu and entering the password to that level, so you then start with your full three lives.

The game overall is well written and has good digitised sound but no background music while you are playing. The sprites are well drawn and move smoothly as does the background.

Quazer also allows you to define the keys by running a separate program from the disc. It creates a character file which is loaded every time the game loads so you have your new key choice every time. The mouse movements cannot be used but the select, menu and move buttons can. Quazer is well worth the £9.50 to members of the Impact Games Club and I recommend it as a good buy. A

The program is contained, along with the 60 crossword files on a 640k disc, and is accompanied by a 12 page booklet. The disc is not copy protected in any way and so a backup is in order. When completing a crossword, clues can be entered as 1A for one across and an option is offered to cancel that selection if desired.

A complete solution to puzzle one is given in the booklet and also a description of how the clue was arrived at. If this sort of help doesn't assist you in completing the other 59 crosswords then 4 levels of computer help are offered. For instance, with each clue is an option 'Help Y/N'. If Y is entered then a further option to mark the clue is provided - useful if you want to mark it to return to later or remind yourself that you have attempted that one.

At level 3 a 'convention indicator' is given. This is a two letter clue - e.g An is anagram. Level 4 is how I how I normally complete my attempts. The first letter of the answer is given at this level and by going through all the clue like this I occasionally fill in a complete word! I did admit to being poor at crosswords, remember? It also possible to fill in all the vowels and then go through the likely consonants to assist yourself if you feel the need. If you do this however the computer will not recognise that the word is complete. Attempt the clue again and fill in the answer in one go or the program won't signal that it is complete although all the letters have been inserted.

I have a few reservations about the program and there a few plus points. I would have liked to have been able to save a part completed game. Upper case is the only form accepted and the point about the computer not signalling that it's complete are my main complaints. The good points are that it's possible to look back at clues even though they have been completed and that it's possible to turn the sound off. I like to play the game at night and it's nice not to disturb the children. There is one point I should mention and that is that there is intellectual difference between the Sun and the Times versions of the Crosswords. Overall I think it's good value and worth the money. Nine out of ten! Times Crosswords, Akom Ltd, £16.95 each. A

Times Crosswords

Chris Walker

If you have just spent several hours typing in the Crossword program from Acorn User, you may be interested in 'The Times Computer Crosswords'. They won't allow you to generate your own crosswords but as in the newspaper of the same name, you pit your wits against a compiler - who says in the accompanying booklet that 'they will provide many hours of pleasure'. The software author insisted that I wouldn't be disappointed with my purchase but I would disagree with his choice of the words '... hours of pleasure'. I think frustration is a more apt description, but I confess to being poor at solving crossword puzzles.

Ace Computing



Euclid

**The 3-D modelling
and animation system
for the Acorn Archimedes.**

EUCLID

Here's what the reviewers say:

“...remarkable value. It already matches or exceeds many functions of other visualisation packages—most of which are far more expensive.”

A&B Computing October 1988

“...very useful indeed.”

Micro User September 1988

“...Euclid will be a great success and a useful tool for any future 3D design programs (a standard even).”

Archive July 1988

“...well worth the attention of anyone interested in this field.”

RISC User July/August 1988

EUCLID - Explore a new dimension!

Price: £45 (inc VAT and P&P).

Available by mail order from:

Ace Computing, 27 Victoria Road, Cambridge CB4 3BW.
Or from your local dealer.



Ace Computing

Configuring the Archimedes

Gerald Fitton

The Archimedes uses memory in a flexible way. For example, the screen display is not always located at the same address nor does it need or use the same amount for every application. Apart from the screen, other specially reserved uses for memory are fonts, relocatable modules, system sprites and the somewhat mysterious 'heap'. In future versions of the operating system we may find ourselves with a RAM disc because that too appears on the list generated by *STATUS. On my 440 when I type *STATUS I get the following:-

Configuration status:

Baud	4
No Boot	
Caps	
Data	4
Delay	20
No Directory	
DumpFormat	0
FontSize	12
No Ignore	
Language	4
Loud	
Mode	0
MonitorType	0
Print	1
RamFsSize	0
Repeat	3
RMASize	2
ScreenSize	5
Scroll	
SpriteSize	5
Sync	1
SystemSize	1
TV	0,1
FileSystem	adfs
Drive	4
Floppies	1
HardDiscs	1
Step	3
SoundDefault	1 7 1
Country	UK

The allocations to look at are those mentioned above namely ScreenSize (5), FontSize (12), RamFsSize (0), RMASize (2), SpriteSize (5) and, for the heap, SystemSize (1). Of course, there is 4 Mb available on the 440. After reserving memory for these special uses I have only 3502332 bytes free

so that about half a Mbyte has vanished. You would be hard pressed to do this on a 305! On the 440, these allocations correspond to ScreenSize (5*32=160k), FontSize (12*4=48k), RamFsSize (0*32=0k), RMASize (2*32=64k), SpriteSize (5*32=160k) and, for the heap, SystemSize (1*32=64k): a total of 464k. FontSize is the odd one out reserving only 4k per unit, all the others are measured in units of 32k. On the 310, these sizes are not allocated in blocks of 32k but in blocks of 8k so that, for example, a ScreenSize of 5 on the 310 would reserve only 40k. Different applications need different amounts of space allocated to different functions. The command *Configure can be used to set these allocations.

How *CONFIGURE Works

If on my machine I type in *Configure SpriteSize 0 and press <return> followed by *STATUS, the SpriteSize appears to have been reduced to 0*32=0k. What may be surprising is that I can still enter Artisan, which uses sprites, and I appear to still have 5*32k available for sprites. This is because I still do have 5*32k available for sprites. Even if I press <break>, <shift-break> or even <reset> this situation is not altered. The only way to execute the change of the SpriteSize to zero is to press <ctrl-break> or to switch off. Having done that, if I now try to load a sprite into Artisan I get the error message "No Sprite Memory"; even if I try a sprite size of 1*32k I often get an error message "Not Enough Room", which is less helpful, even if it is true. If you do not have Artisan, try *Configure ScreenSize 1 (440) or *Configure ScreenSize 3 (310) which, on the 440 will give the error message Bad Mode for modes greater than 7 (i.e. any mode using more than 32k) and the same message if you try any mode other than 0,1,4,5, and 6 (which need more than 3*8=24k) on the 310.

What NOT to do

Acorn strongly recommend that software suppliers do not use *Configure followed by a clever software version of <ctrl-break> that you don't see happen. However, there are some applications around where your existing STATUS is read and stored on disc; this is followed by a software *Configure and the software <ctrl-break>, with your old STATUS

being restored when you use the recommended exit from the application. This is risky because, if you just switch off in the middle of your application then the configuration of the software supplier becomes your configuration and you won't realise it!

The Acorn Way

What Acorn recommend is that the software supplier should provide written instructions for the user telling them how to (manually) *Configure their machine so that it reserves the right amount of space for the application. The instructions should also tell the user how to put it back to the original size afterwards. This method is becoming more popular with suppliers so that making enough room for sprites or fonts is not the problem it used to be.

Not Enough Memory?

A problem that still arises quite often is that some applications need all the memory they can lay their hands on. In these cases it is necessary to reduce the amount of memory allocated to things that don't need it. For example, if you are using screen modes that need only 80k then you can cut down the ScreenSize and make room for the application. In that particular case, on the 310 (or 305) you need *CONFIGURE ScreenSize 10 followed by <ctrl-break> and rebooting the program.

An Exception

Unlike SpriteSize, ScreenSize, FontSize and SystemSize, the RMASize is not, as you might suppose, all that you have. RMASize is what you have AFTER entering the application. If, say, your RMASize=64k, this means that you can *RMLOAD as many modules as you like before RUNNING a BASIC program, and then you can still *RMLOAD modules up to a 64k total from within the BASIC program.

The Future

Acorn strongly advise software writers NOT to use what they call the System Sprite Area. This is the area of memory allocated by SpriteSize. They suggest that the application writer defines their own area of memory and use that instead. However Artisan and many other programs do use the system sprite area. Acorn always have a reason for their advise as I have found to my cost. Usually it means that when the operating system is upgraded the old software doesn't work any more without some complicated "fix"!

What to do?

Finally, how do you know what to do if your program fails because you have allocated the wrong amount of space. Generally, if you keep an eye on the program you can spot whether you are trying to use a screen mode that uses a lot of memory, load a module, a sprite or some fonts. With practice you will be able to use *INFO or *EX to discover the size of sprite, etc. that you are trying to load. When it seems that space is short, increase that area with *Configure one unit at a time, follow with <ctrl-break> and then reboot in the usual way. When you need to reduce the amount of memory reserved for special uses, consider whether the application uses sprites, fonts or modules from within the application and then, if they are not used, reduce the size of that memory allocation to zero. I suggest that you do not alter the SystemSize from 32k. Note that a ScreenSize of 0 does not set ScreenSize=0 but sets it to a default value. Zero means zero for other reserved areas of memory. A

Small Ad's

Colour Monitor Original Acorn colour monitor, as new, sale due to up-grade to multi-sync monitor. Offers around £120. Ring Simon on 0532-842780.

I/O Podule with MIDI add-on interface, as new, boxed with all documentation £80. Phone 01-908-1185, preferably evenings.

Software for sale: Minerva System Delta Plus - £35, Sigmashell - £45, Colton Pipedream £85. P Hughes: 01-840-5650 after 6 p.m.

Archimedes 310 for sale, 4 way backplane + fan, 5.25" disc drive, 1st Word Plus, Presenter, as new, will split. Phone 0208-84-402 for details.

Viewstore (Acorn) for the Archimedes in original packaging, £25 ono. (0234-67067)

"Combo" mains powered dual drives (5.25"+3.5") with Watford external drive adaptor: £180. Also, Conqueror game £12.50. Phone John Ward, Epsom (evenings) on 03727-43887.

Minerva Business Accounts for use with System Delta Plus: Sales Ledger, Purchase Ledger, Nominal Ledger, Stock Management, Order Processing/Invoicing. Brand new. Packages listed at £65 each or £280 for all 5, will sell for £175 - purchased in error. Ring Paul on 0603-507057. A

EXPANSION CARDS FOR THE ACORN ARCHIMEDES COMPUTER SYSTEM

IEEE488 INTERFACE a full implementation of the standard for automatic test and measurement systems

16 BIT PARALLEL I/O two 16 bit input or output ports with handshake lines for digital control applications

DUAL RS423 SERIAL INTERFACE for communicating with two additional RS423 or RS232 devices eg printers, plotters, instruments, etc

12 BIT ANALOGUE I/O in development

All the above high performance expansion cards are supplied with high level software for ease of use and a comprehensive user guide.

Take advantage of Intelligent Interfaces' expertise and purchase a complete Archimedes Computer System.

Officially appointed Acorn Scientific Dealer.



Intelligent Interfaces Ltd

43b Wood Street
Stratford-upon-Avon
Warwickshire
CV37 6JQ

Tel: 0789 415875
Telex: 312242 MIDTLX G



Dabs' Archimedes BASIC Compiler

Brian Cowan

I had come to think that a good BBC BASIC compiler existed only in the realms of fantasy. Imagine then my excitement when I discovered that not one but two such compilers were soon to be available for the Archimedes, one from DABS Press and the other from Silicon Vision. I received a pre-release copy of the DABS press one, and I can tell you now, I am very impressed. I will report on the other one as and when it arrives.

Who wants a compiler?

First, let's look at why you might want a compiler. Most languages operate in compiled form. That is, the whole program is converted (compiled) into machine code before it is run. The machine code program is loaded into memory and run whenever it is required. BASIC usually operates differently. The program is stored as lines of BASIC commands and these are converted to machine code line by line as the program is run. This is known as an interpreted language.

The obvious benefit of a compiled language is speed of operation; compilation to machine code is done prior to running. Each time an interpreted language program is run it must be translated to machine code line by line as it proceeds. The commonly quoted disadvantage of a compiled language is the large size of the compiled program and therefore the memory required; the volume of compiled code is substantially greater than the original source code, whereas a BASIC program, stored as tokens representing each keyword, is considerably smaller than its corresponding text representation. Note however that the interpreter program must also be present in memory when interpreted programs are run.

From the above considerations we can appreciate the popularity of interpreted BASIC on the early micros with their incredibly limited memories. Also, Acorn has consistently excelled itself in the speed of their BBC BASIC. It could be argued that with modern machines, particularly those such as the Archimedes, the above arguments no longer apply. After all, Archimedes has both phenomenal

speed and a huge memory (at least by yesterday's standards). Although speed advantage can still be important, there must be other aspects to the compiler - interpreter debate.

Benefits of interpreted languages

It is far easier to develop programs in an interpreted environment; you can type in a few lines at a time and test them immediately by running the program. This is inconvenient and time consuming with a compiled language, although Turbo Pascal has made considerable advances here. Also, an interpreted language allows easy "on site" program maintenance and modification; you check and/or modify the actual program in the computer. With a compiled language you must find the original source program which may be lost or even purposely destroyed. If any changes are made to it then recompilation is necessary, so you must also find the compiler for the particular computer.

Have your cake and eat it!

Turning to the question of a BASIC compiler, here you can have the best of both worlds if the compiler is up to standard. Programs may be developed in the interpreter environment where they can be tested to see if they work properly. The program can then be compiled for speed increase or for security.

There is a further use for compilation. Even if you intend to run a program in interpreted mode you can run it through the compiler just out of interest. The compiler tests every line, even infrequently or unused ones. Thus it may well pick up bugs that have been overlooked in ordinary testing.

The ABC package

The DABS Press ABC or Archimedes Basic Compiler comes as two discs and two books. One disc is the ABC compiler and the other contains various example programs. There is a user guide of about sixty pages and a reference guide of some 150 pages. I was using a pre-release version which means that the few bugs I encountered should hopefully have been fixed by the time ABC hits the streets. It is interesting to note that ABC was written by an ex-employee of Acorn, who must know the

Archimedes inside out. Paul Fellows was head of the team that wrote the Archimedes Operating System, and prior to that he was head of the Languages Team, so one has high expectations of this product. Apparently the compiler was written in BBC BASIC and then used to compile itself!

Documentation

I was very impressed with the quality of the user guide, and the reference guide was an absolute joy to read; certainly up to the extremely high standard of DABS. Following recent books from DABS they do unfortunately seem to be rivaling the Guardian in the misprint stakes. However I am happy to report only a few errors in the (pre-release) documentation. Chapter one of the user guide is called "Getting Started". This was for me! It explained that a working disc must be produced for serious use, but that an introductory demonstration can be done using the compiler disc directly – a very "user friendly" approach.

Installation

To use ABC, it must be installed either on a floppy disc or on a Winchester. The guide tells you to run the configuration program and that you will be prompted to change discs as required. The default for the installation was a Winchester, so that various things on the screen had to be altered. The manual is clear and there were no problems thus far, but running the program got me into a bit of trouble. The status window kept on throwing up error messages such as "wrong disc" and "please check disc". Maybe I am just plain thick, but this was rather annoying, particularly so, when installation on a Winchester went like a dream. Then it dawned on me... Those messages, which seemed to me like statements about errors were actually what the manual referred to as 'prompts to change the disc'. It is transparently obvious to me now and it was obvious to the writers of the software, but it was not obvious to me as a novice with no experience of the system. (I understand that the release version may have a more user-friendly set of prompts!)

Testing it out

As you will have guessed, I did get the thing installed on a floppy disc and I was keen to try it out in earnest. There are various benchmark tests which

DABS have done and I will discuss those below. I have been using J. Phelan's pi program. This was the one I used in testing the ROM speedup obtained by reprogramming the MEMC (Archive vol 1 no.11). I saved the BASIC pi program to my working ABC disc and started up the compiler. It worked perfectly first time. But then came the moment of truth. Would the compiled code run, and if so, how fast?

I ran the program and decided on trying for one thousand digits. It zapped through in 51 seconds. (The uncompiled BASIC program calculated the same number of terms in 202 seconds and with speeded up ROMs, it took 151 seconds.) For those who are interested, I have included a copy of the compiled pi program on this month's program disc. Don't forget to install the floating point emulator module (or co-processor !!) otherwise the compiled pi program won't run. Floating point is not used in the calculation of pi, but in calculating the time spent doing it!

Exceptions

I tried some other BASIC programs and most compiled without hitch but some did not. The compiler will not support array assignments or passing arrays to procedures. The reference manual explains: "The compiler supports all of the array handling features of BASIC IV. It provides neither the extended features of BASIC V for handling whole arrays nor the local array handling. Other restrictions are that array elements may not be used as the control variables of FOR loops and the maximum number of dimensions an array can take is eight. These restrictions arise essentially from a compromise, to keep the size of the compiler down."

There is the further restriction that multiple exits from structures (FOR loops, REPEAT..UNTIL, WHILE..ENDWHILE and procedures) is forbidden. This forces programmers to structure their programs correctly. (Or at least I think it does, except that the way to modify existing programs is to replace the extra exit statements with GOTOS!!) The EVAL keyword is not supported. This is hardly surprising since EVAL provides a function that is not determined until run time. Compilation of this is therefore not possible; the compiler would have to include its own BASIC interpreter!

Floating Points

In creating a compiler for BASIC, the writers must decide how to implement floating point operations. In fact Acorn faced the same decision in producing the BASIC interpreter. Either floating point support is included in the package or the floating point coprocessor/emulator is invoked. Interpreted BASIC adopts the former approach. This may seem surprising, but there were good reasons. It is a question of precision and standards. BBC BASIC has always used five bytes to store a floating point number. This is a greater accuracy than most micros, and it is also more accurate than the standard single precision, as implemented on the floating point emulator/coprocessor. These conform to the relevant IEEE standard, using four bytes for single precision and eight bytes for double precision. Using five bytes, interpreted BASIC V has then to include its own floating point routines. The ABC compiler uses exclusively standard single precision. Thus it is able to use the floating point emulator/coprocessor, but at a cost of some accuracy. There is also some cost in speed until the floating point coprocessor is available. The present version of ABC does not, unfortunately, support double precision.

Assembler

I was rather surprised to discover that ABC supports fully the BASIC in line assembler. It is a little crazy to compile programs which include assembler since the actual assembly to code takes place at run time. The reference guide explains that it is better to do any assembly separately as external routines and then call them when required. The assembler is included in the compiler solely for the purpose of maximising compatibility with interpreted BASIC V.

Relocatable Modules

An attractive feature of the compiler is the facility to produce relocatable modules. The operation of the Archimedes relies heavily on modules, which were actually the brainchild of Paul Fellows. The current versions of all Acorn compiled languages for the Archimedes are not capable of producing modules. It is expected that later releases, particularly of C will rectify this. However, we have a long time to wait. Until now the only way to produce a module was to write it in assembler. This

is difficult, particularly if you want to use floating point calculations. But now the ABC BASIC compiler can be used. The operating system can in this way be extended and you can implement your own star (*) commands. All three types of module: Application, Utility and Service, can be produced. ABC includes a straightforward procedure for producing the module header. The module facility is a most useful part of the compiler system.

Applications disc

There is an extensive range of demonstrations on the applications disc. Many of these demonstrate the increase in speed obtained using the compiler. There is a ray tracing program and other graphics and even sound programs, and there are some simple relocatable module examples, one of which is a CMOS RAM utility. There is a readme file on this disc which gives the latest updates to the manual together with explanations about all of the programs on the disc. Also there is a fairly intelligent disassembler included. And finally there are the benchmark tests.

New PCW Benchmarks

The applications disc contains quite a number of the standard speed benchmarks. This is a very useful inclusion since the programs are in BASIC as well as compiled code. Thus they could be used with other compilers etc.!! I include the results below, which make interesting reading.

Benchmark	BASIC V	ABC	Ratio
GRAFSCRN	1.68	0.84	x 2
STORE	6.50	6.50	1:1
TEXTSCRN	2.46	2.23	1:1
INTMATH	8.19	0.92	x 9
REALMATH	0.26	0.30	1:1
TRIGLOG	1.20	3.38	x 0.35!!!

Sieve of Erastothenes (1651 primes)

SIEVE	5.16	0.58	x 9
-------	------	------	-----

Recursive Functions

TAK(18,12,6)	27.53	0.78	x 25
Fibonacci	49.43	1.40	x 35
Ackerman(3,4)	4.89	0.12	x 40

Array Operations

Int-Array	1.84	0.34	x 5
String-Array	2.40	1.60	x 1.5
Real-Array	2.04	1.34	x 1.5

Integer Loops

WHILE	13.11	0.40	x 33
REPEAT-UNTIL	12.75	0.37	x 34
FOR-NEXT	2.15	0.29	x 7

Perhaps most remarkable is the TRIGLOG test where the compiled code runs three times slower. All floating point operations here use the floating point emulator. It is a tremendous tribute to the writers of the BASIC V interpreter that the floating point calculations are both so accurate and so fast. The compiler will benefit here from the floating point coprocessor. In most other areas the compiler gives considerable speed increases. This is particularly so for integer arithmetic (one of the reasons for my interest in calculating pi). Recursion and loops are speeded enormously. This is to be

expected since here an interpreter is interpreting the same lines many times.

Conclusion

This is a superb package which I thoroughly recommend to all those who require a BASIC compiler. But remember that Silicon Vision will be launching their compiler soon, so you may wish to wait until a comparison can be made. ABC is a well designed package which is straightforward to use and the quality of the documentation, the two guides, is up to DABS usual high standard. Assuming the bugs I encountered are fixed on the release version, my only reservations relate to the lack of BASIC V array facilities and the lack of double precision arithmetic. A

BASIC V Forum

Clifford Hoggarth

GETting IN KEYpresses

Many programs require the ability to examine which keys on the keyboard have been pressed by the user and with the Archimedes there is also the question of the mouse buttons to consider. BASIC V provides a few different methods of interrogating the keyboard and via the easy access to operating system routines using the SYS command, a few extra facilities are available.

Interrogating the Keyboard Buffer

The BASIC functions GET, GET\$ and INKEY return information about character input. GET\$ and INKEY\$ return single character length strings while GET and INKEY return the ASCII value of that character. GET and GET\$ wait for a keypress, whereas INKEY and INKEY\$ require a positive numerical parameter which is the length of time in centiseconds for which BASIC (strictly speaking the O.S.) will wait for a keypress. Hence:

```
char=GET  
char$=GET$
```

wait for an infinite time, whilst

```
char=INKEY(100)
```

waits for one second (100 centiseconds), and

```
char$=INKEY$(0)
```

will not wait for any time at all!

If the INKEY function reaches its time limit then the function returns the value -1. INKEY\$, if the time limit is reached, simply returns a null string.

An important thing to realise is that these functions actually examine the keyboard input buffer and not the keyboard directly. This means that any previous, but as yet unused keypresses which have been buffered will be used first. Try the following to see this effect.

```
FOR delay=1 TO 10000  
    PRINT TAB(0,0) delay  
    NEXT delay  
REPEAT  
    PRINT GET$  
    UNTIL FALSE
```

If you type a few characters while the FOR...NEXT loop is counting then you will see these characters printed out after the loop reaches 10000 as they have been stored in the keyboard buffer. (N.B. you will have to press <escape> to leave the REPEAT...UNTIL loop).

Now if you wanted to ask the user of your program to press a key to move onto the next part of the process or to make a selection, or to confirm an action before it is carried out, then, if the buffer contained keypresses, the key returned may not correspond to the desires of the user. At worst, valuable data could be lost if e.g. a file was deleted

because of this. If it is important that it is the NEXT key pressed after the function is called then the buffer must be cleared (or "flushed") first. This can be done using an OS_Byte call to the O.S. i.e.

```
SYS "OS_Byte" 15,1
```

which flushes the "current input buffer" (which may be the RS423) or,

```
SYS "OS_BYTE" 21,0
```

which flushes the keyboard buffer specifically.

Try inserting one of these command lines before the REPEAT in the example above and running the program again.

Forgetting to flush the buffer when necessary can be the cause of many problems and bugs in some programs. It is also infuriating when a program merrily pages through several options because you accidentally lent on the keyboard!

Because of the buffer, defining a function key string can make one keypress appear as several, indeed this is how the function keys work – the O.S. puts the characters of the string into the keyboard buffer.

Non-ASCII Key Values

The above functions return values based on ASCII characters. Try something like:

```
REPEAT
  PRINT GET
  UNTIL FALSE
```

to see what values different keys produce. Note that <shift> and <ctrl> affect the values produced.

You may have noticed that some keys did not produce any values, namely the cursor keys and the special keys above them. This is because these keys have no ASCII character equivalents. So how can these be detected?

The cursor keys can be made to behave as character keys using another OS_Byte call:

```
SYS "OS_BYTE" 4,1
```

This makes the cursor editing keys return the following values:

Copy	135
Left arrow	136
Right arrow	137
Down arrow	138
Up arrow	139

Note that the Page Up and Page Down keys act like the Up and Down arrow keys respectively.

To restore the cursor keys to normal operation, use:

```
SYS "OS_BYTE" 4,0
```

Examining the Keyboard Directly - and the Mouse Buttons

INKEY can also be called with a negative number as its argument. In this case the value identifies a specific key on keyboard and the function returns a TRUE (-1) or FALSE (0) depending on whether or not that key is pressed. The important difference to note here is that the function now examines physical KEYS rather than the characters in the keyboard buffer. The side effect of this is that you cannot distinguish between "b", "B" or "<ctrl-B>". However all the keys on the keyboard can be examined including the special keys cursor keys, and <shift>, <ctrl>, <alt>, etc. keys, and also the mouse buttons.

Each key on the keyboard has a negative INKEY value associated with it, which can be found in appendix F of the User Guide (not E as stated in the text relating to INKEY). From this list it can be seen that it is even possible to distinguish between e.g. the left and right <shift> keys, and the numeric keypad keys are different from those on the top row of the main keyboard area.

This form of the INKEY function can be more difficult to use if just characters are needed, and is for testing for a specific key press at that moment. It can be used for monitoring the holding down of key e.g.

```
IF INKEY (-99) THEN REPEAT UNTIL NOT
INKEY (-99)
```

tests the space bar and, if it is pressed, waits until it is released. (The NOT means that the UNTIL will only be TRUE if INKEY(-99) is FALSE, i.e. the key is not being pressed). This could be used for pausing the output of a program, for example.

The other major use, is for monitoring multiple key presses. For example in a program pressing an arrow key may move a cursor, but if a <shift> key is pressed then the cursor moves faster. The following demonstrates this by printing the new position of the cursor.

```

position = 0
REPEAT
    dirn = 0
    IF INKEY(-122) THEN dirn = -1 : REM left arrow
    IF INKEY(-26) THEN dirn = 1 : REM right arrow
    IF INKEY(-255) THEN dirn = dirn * 8 :REM <shift> key
    position = position + dirn
    PRINT TAB(0,0) position
UNTIL FALSE

```

Monitoring keypresses in this way is obviously the best method for use in games, etc. when you do not want the keyboard repeat rate to have any effect (this actually controls the rate at which repeated characters are placed in the keyboard buffer).

Negative INKEY values also exist for the three mouse buttons, enabling them to be treated as keys, rather than using the MOUSE command. The relevant numbers are:

Select	-10
Menu	-11
Adjust	-12

So to allow e.g. a selection to be made by pressing return or <select> then the test

INKEY(-10) OR INKEY(-74)

will be true if either are pressed.

Function Keys

If you tried the earlier listing which showed the values produced by various characters with GET you may have noticed that the function keys alone did not produce any values (assuming they had not been defined) but <shift> + function key did. It is possible to alter the values returned using further OS_Byt calls. However this subject is further complicated by the use of <shift> and <ctrl> keys with function keys and that there are two different groups of function keys which include the cursor editing keys. Hence I shall just mention that more flexible handling of these keys is possible and refer anyone interested to the relevant section of the PRM. (Similar controls can be placed upon the numeric keypad).

Hopefully this has shown how key input can be monitored in different ways, the choice of which

you use is dependent on the particular needs of the program. I think the best advice I can give about how to select a method is to use the simplest one that works. "Simplest" here means the easiest to program, so although negative INKEY values may initially appear more complex, the actual program code may be simpler to implement.

For those of you really keen on investigating the possibilities, reading the PRM will reveal several other ways of controlling and examining the keyboard using further O.S. calls, e.g.

```

SYS "OS_Byt" 219, newvalue
causes the TAB key to generate the ASCII character
newvalue. If newvalue > 128 then the TAB key can
be made to mimic one of the function keys.

There are not many examples above as a large
number would be necessary to demonstrate all the
possibilities, and the best way to understand the
differences between the functions is to try them out
for yourself. I will however leave you with the
following short program which uses all of the
above, without actually doing anything useful!

PRINT "Press SPACE to start"
REPEAT UNTIL GET = 32
FOR delay=1 TO 5000:NEXT :REM delay to
allow normal release
IF INKEY(-99) THEN
    PRINT "Please release SPACE bar"
    REPEAT UNTIL NOT INKEY(-99)
    PRINT "Thankyou"
    ENDIF
PRINT Wait...
REPEAT UNTIL RND(10) > 8
PRINT "Press any key"
SYS "OS_Byt", 21, 0
key$=INKEY$(100)
IF key$<>" " THEN
    PRINT You pressed key$
    IF INKEY(-1) THEN
        IF INKEY(-4) THEN PRINT "and the
left shift key"
        IF INKEY(-7) THEN PRINT "and the
right shift key"
        ELSE PRINT "and no shift key"
        ENDIF
    ELSE PRINT "Too slow"
    ENDIF A
ENDIF A

```

Now available

£105 from Archive

PipeDream

PipeDream is now available on the Acorn Archimedes. It provides comprehensive word processing, spreadsheet and database facilities integrated in a way only dreamed of before.

With other integrated packages, you have to divide your work into artificial sections, such as text, numbers and calculation, and database.

With PipeDream, you compose your work in the order you want to print it, with text and numbers all together in one document. Incorporate calculations directly into paragraphs of text and formatted paragraphs directly into spreadsheets.

PipeDream is ideal for all tasks involving words and numbers. From writing thank-you letters to encyclopaediae, invoices to cash flow forecasts, stamp-collection records to inventory management, film scripts to mail-shots.

PipeDream is 100% file and keystroke compatible with Z88 PipeDream and PipeDream on the IBM PC. It is also compatible with View Professional on the BBC microcomputer. You can create documents on any of these computers, transfer the files to any other and continue working, with no loss of information. No other software enables you to share your files with all these computers.

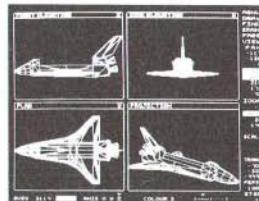
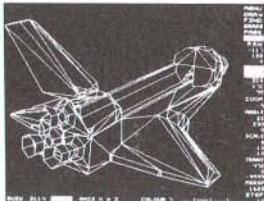
PipeDream for the Acorn Archimedes costs £99 + VAT.

It is not possible to detail all of PipeDream's facilities here. For full details or to order PipeDream call us on 0954 211472 or write to us at Colton Software, Broadway House, 149-151 St Neots Road, Cambridge CB3 7QJ.

PipeDream - power at your fingertips.

SILICON VISION

SOFTWARE FOR THE ARCHIMEDES & BBC

SolidCAD

The ultimate 3D Drafting System for Architectural Design, Interior design, Engineering Design and Teaching CDT. Allows drawing in plan, front & side elevations and also directly in 3D view. Includes powerful zoom & pan options for precision drafting and surface definition for creating solid colour objects. Also includes Sweep, Extrude & Macro facilities for designing very complex objects easily. Designs created with SolidCAD are compatible with the Realtime Graphics Language for high-speed flicker-free animation. The custom Archimedes version also performs smooth shading for realism.

SolidCAD/Arc users can upgrade also to the Realtime Solids Modeller (Arc) for £40.00.

£49.95 (ARC & BBC) New

REALTIME SOLIDS MODELLER

This package includes both the sophisticated design environment of SolidCAD and the high speed animation capability of a Realtime Graphics Language (RGL) module developed in pure ARM Risc code for supercharged performance. The package is ideal for Architectural design, Interior design, Engineering design & teaching CDT. The RGL module can be used to create standalone flicker-free animation of designs from your own programs. Smooth shading is also performed for realistic images. Through our in-house expertise in 3D Design and High-speed techniques, no other package can rival the design environment & animation speed of the Realtime Solids Modeller.

£89.95 (ARC) New

REALTIME GRAPHICS LANGUAGE

The Realtime Graphics Language ROM provides a complete 3D Solids Wireframe animation system with 52 star commands and 3D Editors for designing objects to animate from your own programs. Includes a 30,000 pixels/sec line generator for fast 3D drawing rates, ID Rotate, Scale, Orbit, Perspective and Turtlegraphics. Also compatible with designs created with SolidCAD (BBC).

£15.95 (BBC)

SUPER-DUMP

This ultimate printer driver which takes advantage of the highest resolution capability of ordinary Epson compatible printers to provide 1920 x 1024 resolution. Images can also be sent, processed and previewed before printing. Fully compatible with SolidCAD, Realtime Graphics Language, Gate Array design system & 3D CAD Animation system. Your own graphics programs or other CAD packages can be made compatible with Super-Dump by the addition of a few simple commands. An example program is included in the ROM.

£15.95 (BBC), £24.95 (ARC) New

Presentation Manager

The Presentation Manager is an integrated software environment to create, edit and play-back computer generated multimedia sequences and demonstrations. Also handles graph plotting for presentation purposes. A full menu system which can be incorporated within the sequence.

£34.95 (BBC), £49.95 (ARC) New

SILICON VISION LTD, SIGNAL HOUSE, LYON ROAD, HARROW

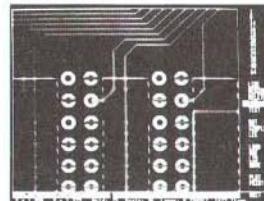
MIDDLESEX HA1 2AG. TEL: 01-422 2274 or 01-861 2173

FAX: 01-427 5169. TELEX: 918266 SIGNAL G.

(Access/Mastercard/Eurocard accepted)

All prices include VAT and Carriage (Overseas orders add £4).

SEE US ON STAND 34 AT THE MICRO USER SHOW

ARC-PCB

The ultimate PCB design system developed specifically for the Archimedes with a specification that cannot be matched. Includes Automatic Routing, Rats-nesting, 8 layers, Surface mount capability, 0.001" resolution, 32 x 32 maximum board size, On-line Help, Fast Zoom-Pan Redraw, Text & Silkscreen facility, Variable Line Part Text Grid sizes, Part Libraries, Block Move, Copy, Rotate, Mirror-Erase options, and up to 300,000 components. For hardcopy, the system supports a large number of plotters and ordinary Epson compatible printers at very high resolutions (1920 x 1024) of 240 dots/inch for near laser quality output. A memory level ARC-PCB system (without auto-routing) is available for £99.95 (Auto-routing upgrade £100). Enquire about our turnkey PCB design systems complete with Colour Archimedes and/or plotters with prices from £1220 to £3909 (inc. VAT).

£195.00 (ARC) New

RiscBASIC

Supercharge your Archimedes Basic programs by compiling them automatically into pure ARM Risc code with the RiscBASIC compiler. Features include Relocatable modules, Cross reference of all variables, functions, and procedures, Floating point and Integer support, Stand alone code generator, Optimising compiler & Full runtime error handler.

£99.95 (ARC) New

RiscFORTH

A new 32-bit implementation of the FORTH 83 standard designed to take full advantage of the ARM architecture. Features include Multi-tasking, Optimising compiler, built-in ARM assembler with floating point mnemonics, built-in Full screen Editor, File system interface, OS calls support, Floating point & Integer maths, WIMP support, Single step debugger, Shadow screen for documentation, Block manipulation, Dictionary & Vocabulary display, Call finding and a stand-alone code generator.

£99.95 (ARC) New

BBC Compatibility List

Richard Averill

Richard has prepared a list of programs from BBC microcomputer / Master and Compact that do or do not work on the Archimedes, with or without the 6502 emulator.

Applications that run under Archimedes BASIC V

Name	Supplier (Comments)
Educational	
A Level Maths/Stats	AVP (Does not transfer all files)
A Level Statistics	BBC soft
Chemical Analysis	Acornsoft
Chemical Structures	Acornsoft
Chemical Simul'n's	Acornsoft
Gas Diffusion	Longman
GCE Biology	Acornsoft (Needs char conversions)
GCE Maths – 1,2	Acornsoft (As above)
GCE English	Acornsoft (As above)
German Master 1,2	Kosmos
Graphs	Michael Jay
Graphs + Vectors	Salamander
M/f of Sulph'ic Acid	Longman (Needs disc deprotection)
Mega Maths 1,2	LCL (Needs char conversions)
Statcalc	Macmillan
Which Salt	Micro Power
Graphics	
Body-Popping	Beebug
Creative Graphics	Acornsoft
Graphito	Addison-Wesley
Graphs + Charts	Acornsoft
Zoom	Beebug
Business software	
Order processing	Acornsoft
Purchasing	Acornsoft

Applications requiring the 6502 Emulator

Educational

Graphic Calculus III	Glentop
Symbolic Calculus	Maths Workshop (Version 1.0 runs with BASIC 2)

Graphics

ViewPlot	Acornsoft (Run conversion program)
----------	------------------------------------

Printer Utilities

FontAid	CJE Micros
---------	------------

Roms

Forth	Acornsoft
Pascal	Acornsoft
VidiEditor	BBC
View	Acornsoft
ViewSheet	Acornsoft
ViewSpell	Acornsoft

Programs that do not seem to run

Graphics

NovaCad	Technomatic
---------	-------------

Games

Most games

Peter Sykes also offers two programs that work in BASIC V with slight modifications: LabelMaster (remove all occurrences of *FX154) and Fax*File Organiser (remove the *LOAD screen command and, just before the REPEAT in FNmenu, add a line with PROCdelay(7)).

If you have programs that you know work on the Archimedes or that only need slight mods, send the information (on 640k disc, preferably) to Richard Averill, c/o Archive Magazine. **A**

Knitware – An Introduction to Systems Programming

N.Harris

Knitting is a cheap, personal way of producing clothes. The results are tailor made and often imaginative, always alterable should you happen to grow and usually better than ones you can buy in the shops.

Knitware (not be confused with Knitwear) is a form of computer program. The term applies to utilities written by individuals for their own personal use. Knitware can be written in the absence of satisfactory Software, or while waiting for Vapourware. This article deals with one aspect of Knitware: systems programming.

In essence, a system is a program that can be used by other programs. A hypothetical example of this would be a speech system which could read out any words that were fed into it. Such a system would enable blind people to use wordprocessors or could act as a voice for those with speech difficulties. These applications hint towards the overall objective of this subject: **Make it easy for everyone else.**

Usually, the traditional attitude of programming is adopted while trying to meet this. Unfortunately it often fails to achieve the objective by its very nature. It can be called the 'something for nothing' attitude, which attempts to make the system work using as little memory as possible.

The opposite attitude is more suited to the Archimedes. It can be called the 'machine sacrifice' attitude, which attempts to make the system fulfil the objective at the penalty of resources. An example of this is shown below.

A graphical user interface, or WIMP system, should be consistent, flexible, and involve good presentation. The design of it can be inspired by existing products, but care should be taken not to copy across faults. The key element of a WIMP is the window. By displaying several windows at once information can be assimilated faster, and can be cross-referenced more easily. Scrolling allows for different parts of their interiors to be seen through the window. They can be presented lying on top of one another and dragging allows them to be

rearranged should neatness require it. Finally, their size can be altered to adjust the amount of interior they display.

Try the Arthur Desktop then read the following.

It is nice if dragging and re-sizing can be done on the actual windows rather than a ghost and that they stay in their positions in the pile until told to come to the front, or go to the back. Also, it is sensible to have all the 'controls' in one place.

The updating algorithm is the most important and difficult part of the system to implement. So it is a good idea to do a feasibility test first. I did mine in five minutes which indicates the algorithm is pretty simple; here it is:

Keep track of all the windows in the pile by number

Reserve a whole screen bank for each window interior

Set aside a screen each for display and workspace

- 1 Clear the workspace screen every update frame
- 2 Work from the bottom of the pile up, drawing windows
- 3 First draw a frame of the correct dragged size
- 4 Interact with the relevant window's screen bank
- 5 Cut a sprite of the correct scrolled size
- 6 Interact with the workspace screen again
- 7 Paste the sprite inside the frame, and repeat 2-7
- 8 Swap over the workspace and display at Vsync
- 9 Interact with the user and then go back to 1

Even though this cut-and-paste method is simpler than the Arthur WIMP, a demonstration program would not go amiss. So there is one on the monthly program disc to clarify matters. There is the possibility of a follow up article about the 'IM' part of WIMPs in the future. Hopefully this article has shown that systems Knitware as seemingly complicated as the WIMP are easy to do provided you have the correct attitudes. **A**

(These references to 'Knitware' remind me of the first young lady who did some typing for me who, from dictation, came up with 'softwear'! Ed.)

LISP – A Wider View

Simon Brooke

In September's Archive, David Wild wrote a short and dismissive review of Archimedes Lisp. It seems that David is not a 'Lisper' and is not very familiar with the language. This is a pity, because I feel that he has badly served what is in fact a potentially excellent product.

This is an attempt to redress the balance a bit; but I don't want it to read as a whitewash. There are some serious bugs in the system and Acorn seem to be making less than serious efforts to correct them.

The problems

So I'll start by talking about the problems. Firstly, producing hard copy from Lisp is (apparently) impossible. Neither <ctrl-b> nor (vdu 2) cause output to be sent to the printer. This, in itself, makes serious application development difficult – who is going to buy an application which can't print? The only (apparently) available option is to write output to a file and print that file later. Not good. (*But see David's comments below. Ed.*)

Secondly, errors generated while output is directed to a file-stream are sometimes not recoverable. The system may appear to recover, and limp on for a while, but will generally die at the next garbage collection. So it is pretty important to ensure that code which writes to files is solid.

Similarly, operating system SWI calls can leave dangerous garbage around, causing LISP to die at the next garbage collection. This is in fact very easily cured, but the cure isn't documented in the manual, so I've described it at the end of the review.

Thirdly, under the WIMP, characters typed at the keyboard cause an undocumented Wimp_Request code (reported to LISP as -647) to be returned; I have not yet found a way either of persuading the WIMP to return the correct request code, or of interpreting the spurious one. (I think this is an Acorn bug, and not one of mine – but one of the things for which I was blaming them for months turned out to be my own fault!)

When is a bug... ?

As for David Wild's supposed (-2 3) bug, I don't believe it is a bug. What happens when you try to

evaluate this expression, is that LISP will recursively try to find a function definition in '-' and it won't find one, because there isn't one (unless you define it). Consequently, it will continue doing this until it reaches the end of the stack (several hundred thousand frames) and then it will print several hundred thousand error messages, saying that it couldn't find the definition. Then it will come back to the LISP prompt. I know it will, because I've just tried it. I can understand David being surprised at this, but it is emphatically not an error and there is no reason to press <ctrl-break>.

Poor documentation

Some of David's other criticisms are just, however. The documentation is ridiculously meagre. Only 375 of the more than 500 functions supplied are described – and many important and useful ones are omitted. Worse, some useful functions mentioned in the manual are not present in the implementation (gcdaemon – a user function called after each garbage collect – is a case in point). Again, as he says, the break package is hopelessly under documented – so much so that I effectively ignore it, except for very obvious things like examining bindings on the stack.

More problems

There are a couple of other problems which David doesn't mention which make life difficult, too. Firstly, although there is a very good in-core structure editor, the system as provided by Acorn has no provision for the saving of source code developed in-core. (There's also no provision for documenting code developed in-core – not so serious, but still annoying.)

The second problem compounds this. If you attempt to save the core image, using the function 'preserve', and a filing system error occurs – on a single disk machine like mine, for example, a disk full error is quite likely – the situation is not recoverable. All your current development work – if you have not separately saved the source – is lost. Worse, if you have in the meantime created any modules, your system disk may be corrupted. I now find saving a core image so stressful that, when I

feel the need to create a new basic system, after loading everything I want into it, I type (preserve) and immediately go away and have a coffee, and don't come back to see if it has worked until I've calmed down!

So the first thing you will have to build yourself is a file package, and I've stuck a simple one on the end of this article. (*Rather too long to print, I'm afraid, so either send an S.A.E. for a listing, or get the monthly program disc. Ed.*)

Finally, on the debit side, Cambridge LISP (of which Archimedes LISP is a good, standard implementation) suffers from a fault which is common to most of the older LISP implementations: its interpreter handles variable bindings dynamically, but its compiler handles them statically. It's a pretty abstruse point, and unless you are doing subtle things it doesn't matter much. But there are times when you will find that something which works perfectly well interpreted won't run compiled – and that's irritating.

Now the good news...

So what's good about it? Well, firstly, like any Archimedes language, power and speed, of course. For example, a Xerox 1186 Daybreak workstation, a specialist LISP machine costing around £20,000, can calculate the factorial of 1000 in 22 minutes and 8 seconds. An Archimedes, running Archimedes LISP, performs the same calculation in 11 seconds – one hundred and twenty times faster.

(Incidentally, Pascal programmers may like to know that the whole code to do the calculation is as follows:

```
(de Fact (n)
  (cond ((zerop n) 1)
        (t (times n (Fact (sub1 n))))))

(Fact 1000)
```

It is of course possible to write a Pascal function to calculate arbitrarily large numbers – but try it. And then see how many man hours it took you. And then cost your time at say £5 an hour. And then tell me whether £200 is a lot to pay for that sort of expressive power.)

Of course Archimedes LISP isn't such an extensive or such a friendly LISP environment as those

provided by the big LISP workstations. But it makes up for those deficiencies by being clean, elegant, cheap – and blindingly fast. There are bugs in it – but not many, and none of them disastrous.

Making the SWI interface safe

As I've already said, there is a risk that making calls to the operating system using SWI will leave garbage which the garbage collector can't digest. Arthur Norman, the author of Cambridge LISP, says:

"if the values are not valid Lisp quantities (e.g. the simple rule is that if the top byte is zero you are OK, otherwise things may be dodgy...) you can crash Lisp if you leave them around long enough for a garbage collection to happen"

The simple cure (suggested to me by Arthur Norman – I merely pass it on) is as follows. Values are passed to the SWI call in a vector. After the SWI call, the vector must be made safe, and this can be done with the following function:

```
(de VectorMakeSafe (vect)
  (dotimes
    (i (upbv vect) vect)
    (putv vect i (ilogand (getv vect
      i) -1)))) )
```

For this function to work, it must be compiled with LISP type checking switched off – and while we're at it, we might as well have another function for finding the actual base address of strings, so that we can pass them to SWI routines:

```
(de StringGetBaseByte (str) (iplus
                                str 4)))
```

Having defined both these, switch off the type checking:

```
(carcheck 0)
```

Compile the functions:

```
(compile '(VectorMakeSafe
           StringGetBaseByte))
```

and remember to turn the type checking back on! Now we can use a SWI call to, for example, read the mouse state:

```
(de GetMouseState nil
  (let
    (val (regs (mkvect 4)))
    (SWI 28 regs)
    (VectorMakeSafe regs))
```

```

(list
  (cons (getv regs 0) (getv
                           regs 1))
  (cdr (assoc (getv regs 2)
               *mouseStates*)))
))

(setq *mouseStates*
      '((0 (1 right) (2 middle) (3
                                   right middle) (4 left)
      (5 right left) (6 middle left)
      (7 right middle left)))
-
```

this function returns a list consisting of the position of the mouse as a dotted pair, and the buttons pressed as a list of tokens. This allows us to write things like:

```

(de GraphSelect (window position
buttons)
  (cond
    ((equal buttons '(left))
     (Window:SetUserItem
      window
      'currentNode
      (Graph:NearestNode
       (Window:GetUserItem
        window 'graph)
       position)
     (Window:Repaint window))
    ((equal buttons '(middle))
     (let
      ((menu (Window:
GetUserItem window 'middleMenu)))
     (cond
       ((and menu (Menu
                     menu))
        (GraphWindow:Recalculate
         window)))) )
    ((equal buttons '(right))
     (let
      ((menu (Window:
GetUserItem window 'rightMenu)))
     (cond
       ((and menu (Menu
                     menu))
        (GraphWindow:Recalculate
         window)))) )) ) )

```

A simple file package

The file package works simply by pretty-printing each of the objects to be filed onto the disk.

Consequently it is rather wasteful of space. Also, it doesn't properly handle either strings or vectors – although it could be simply modified to make it do so – and it won't save circular structures. To save functions called foo, bar, and baz into a file called FooBar, type

```
(WriteFile 'FooBar '(foo bar baz))
```

To reload the file, type

```
(rdf 'FooBar)
```

The file package puts the list of the names of the objects in the file onto an 'objects' property of the filename. Consequently, if you want to make some changes to FooBar, you could load it in using '(rdf 'FooBar)', make the changes, and save it again by simply typing

```
(WriteFile 'FooBar)
```

Or, you can compile the contents of a file using:

```
(compile (get 'FooBar 'objects))
```

David Wild Replies....

I am very grateful to Simon Brooke for his detailed reply to my brief review of Acorn's Lisp for the Archimededs. Although he says that I have "badly served a potentially excellent product" I feel that much of what he says reinforces my view rather than contradicting it.

I must accept that I was wrong in thinking that (-2 3) produced an irrecoverable error, but would plead that the length of time spent watching these error messages fooled me. He accepts, though, that there are bugs – and some of them are potentially serious – and it does seem unfortunate that the language should be sold with a release note warning you not to use one of the functions provided in the package.

At the root of most of the problems, though, is the documentation. Simon tells us that operating system SWI calls can leave dangerous garbage around, and that the cure is not documented in the manual. I wouldn't have discovered this for myself because the SWI calls are not documented either! As you examine the list produced by (oblist) you find there are many other functions undocumented and some of them can be very useful indeed.

As Cambridge Lisp is not the same as any other version of Lisp, this question of documentation

becomes vital if the language is to be successful and attract potential users to the Archimedes. As a minimum, I feel, there should be an illustration of each function actually doing something, even if what is done is trivial. (These illustrations should also be proof-read!) As an example the "do" macro is illustrated by :-

```
(do ((var1 init1 inc1) (var2 init2  
                           inc2)...)  
    (exitcondition resultvalue) body1...),  
which is not very informative for someone who  
does not know the language already. A better  
method might be to have a formal definition  
followed by an illustration such as :-  
(do ((x (oblist) (cdr x)) ((null x) nil)  
     (progn (princ (car x)) (princ  
                   " ") (princ (plist (car x))))).
```

Such illustrations would make it very much easier to understand what the macro or function was intended to do. It is not really fair to refer the programmer to a (very expensive) book on Common Lisp for these details as nowhere is it explained which features of Common Lisp are not used in Cambridge Lisp.

Another aspect of the problem is that if someone like Simon, who has privileged access to the man who wrote the language, cannot find out how to make the system print, what hope is there for the rest of us? As it happens I have managed to solve the riddle, but I suspect that it was beginners luck. I defined two functions as follows :-

```
(de printon () (interpret-  
                 string "fx3,8")  
(de printoff () (interpret-  
                 string "fx3,64") (vdu 3))
```

and found that these worked. The odd thing is, if I leave out the (vdu 3) in the "printoff" function, printing stops while I am in Lisp, but restarts as soon as I go back to the operating system! There must be a suspicion that "illegal" methods are used because my modem

tinkles if it is left attached to the serial port while Lisp is running, and this doesn't happen with any other program.

As I said at the beginning of this piece I am glad that Simon wrote the way he did, because his letter encouraged me to work at using Lisp, and find out many of the things which it can do. I still find it irritating that I have to do so much detective work when a properly written and proof-read manual would have let me try out problem solving of a different kind. While I agree with Simon that Lisp is a potentially excellent product, the present version is rather like giving a blind man a guide dog which has an excellent pedigree but which hasn't been house-trained. If the blind man has to spend much of his time clearing up the puddles, he isn't going to find his way round the town. **A**

(David has also provided a set of very useful utilities which again are too long to print. Either send an S.A.E. or buy the program disk. Ed.)

COMPUTERWARE

SPECIAL OFFERS

CW

- 4 Slot Archimedes Backplane complete with fan etc. £59.95
- 20Mb Hard disc podule £399.95
- Podule less drive £249.95

Offer closes 31st December 1988

All Computerware products come supplied with full instructions and mounting metalwork. Prices shown are fully inclusive UK.

Please allow up to 28 days delivery.
Send S.A.E. for details.

order now from: 55 Romulus Court,
Justin Close, Brentford Dock,
Brentford, Mddlx,
TW8 8QN.

Disc Recoverer

Richard Averill

There are many different ways in which corruption of discs occurs. This program deals with the most common type of disc corruption experienced by Archimedes users.

This occurs when the ADFS does not recognise newly inserted discs and overwrites the catalogue, making the files on the disc inaccessible using the normal commands. However, the ADFS is a well-designed system that will allow your files to be recovered. The program listed below will do all the work for you in restoring your 800k 'D' disc as close as possible to its original state.

What the program can and can't recover

The extent of recovery that is possible depends on how you organise your discs. It is advisable to adopt the recommended approach of organising everything into subdirectories and have as few files as possible in the root directory. This may at times seem tedious but it is the best method of organising your discs, since any files in the root directory will be lost, unless you are prepared to use a disc sector editor and a lot of patience! This program will recover all files on a disc apart from those in the root directory, so if you organise your files carefully it should be possible to recover everything except the !Boot file! If, however, the disc has been overwritten with other files, then you may find some of the files will not contain what they used to!

The root directory data is always stored in sectors 1 and 2 of track 0, with what is known as a 'Boot sector' at track 0, sector 0. If this directory information has been corrupted, it is these 3 sectors that have been overwritten, everything else will probably (hopefully) be as before. The program deliberately ignores these 3 sectors and proceeds with the disc recovery process as if the disc catalogue did not originally exist – it will work just the same on non-corrupted discs as it does on corrupted discs! The rest of an ADFS disc is taken up with directories, files and free space. Due to the way that the program works, you may find that any previously deleted directories will be restored to the root directory.

Who is Hugo?

I don't know who Hugo is, but the presence of this string at the beginning of a sector indicates that it contains directory data. This is how the program finds all the subdirectories. The text 'Hugo' appears at bytes 1 to 4 at the start of the sector that contains subdirectory data. The program reads all data on the disc into memory so that an 800k memory space is required (i.e. a 310 or 400 series machine is necessary). A search is then made through all the 800k of space – until Hugo is found. It doesn't actually look for "Hugo" but for the corresponding ASCII values (&6F677548) to speed the search up. If it finds a 'Hugo', it will write the disc address where Hugo was found and the directory's title to a directory list contained in two BASIC arrays.

When the program has processed all the disc, it then looks at each directory in turn and cross-references all the items to find out which directories originally branched off the root directory. Any lower level directories will be ignored at this stage, a flag being set to true if a directory is found to be such a lower level directory.

Recursion

In the DFS system, all the file information is stored on sectors 0 and 1, but the ADFS is essentially a hierarchical system with directories containing subdirectories and so on. This means that a special recursive procedure is needed to restore all of the disc data.

The program calls the disc restorer procedure and basically goes through each directory in turn writing all of the files in the directories to disc. Eventually all of the directory levels are explored in this way. The recursive procedure PROCExamine is very powerful, and uses LOCAL variables to good effect like a stack, but simpler. There are no OSCLI ("SAVE "+etc.)'s – these aren't meant for programs to use. The recoverer uses official "OS_File" and "OS_FSControl" SWIs to speed things up.

Using the program

Type RUN (what else!) and answer all the questions. Insert your original disc when prompted

Disc Recoverer

and have ready a blank disc for the restored data. The only time the program will write to disc is when you have given confirmation. By giving appropriate answers to the prompts you can format your destination disc or delete files whilst the program is running.

Limitations

Since 640k ADFS discs use a different disc format and catalogue organisation, they cannot be recovered using this version of the program.

The author is currently adapting the program so that it will recover 640k discs. Hopefully, it will be possible to present this program in the next issue of Archive magazine.

Prevention is better than cure

Finally here is some advice about taking care of your floppies:

1. Always take regular backups of your work.
2. Be careful writing to a newly inserted disc that has not been properly *MOUNTed.
3. Organise your discs into subdirectories.
4. Use good-quality branded discs!
5. Make sure that you know about the physical construction of a disc. The metal head-cover on a 3.5" disc of mine containing the latest version of the Archive menu fell off one day, leaving me to figure out how to put it back!
6. Never leave discs around sources of magnetism (e.g. transformers in power supplies, telephone handsets, loudspeaker cabinets, monitors etc.)

```
10 REM > GetDisc
20
30 REM (C) Richard Averill, 1988.
40 REM Recover directories on a disc
disc with its root directory wiped
50 REM Runs on 310 with sizes at 0
(except ScreenSize!), or a 440.
60
70 MODE 12
80 PRINT TAB(6);'"Archimedes disc
recoverer. (C) Richard Anthony
Averill, 1988."
90 PRINT "This program will re-
create a disc that has had its
root ($) directory over-"
"written. Concept by Richard
Anthony Averill."
100
```

```
110 DIM surface% 800*1024
120 DIM notcopy%(150), directory%
(150), directory$(150),
copylist%(150), copylist$(150)
130
140 DiscOp% = &40240
150 hugo% = &6F677548
160 OS_FSCtrl%=&29
170 OS_File% = 8
180 dircounter% = 0
190 copycounter%=0
200 INPUT "Examine which drive ? "
drive%
210 PROCprompt("Insert disc in drive
"+STR$drive%+" and press any key
...")
220 PROCloaddisc
230 PROCFinddirs
240 PROCFindroots
250 PROCDisplaydirs
260 PROCCreatedisc
270 END
280
290 DEF PROCloaddisc
300 PRINT '"Loading disc into memory
...
310 SYS &40240,0,1,drive%<<29,
surface%,800*1024
320 PRINT "Finished."
330 ENDPROC
340
350 DEF PROCCreatedisc
360 INPUT "Create a new disc in
which drive ? " drive%
370 IF FNask(CHR$10+CHR$13+"Drive
"+STR$drive%+. OK ?")=FALSE THEN
ENDPROC
380 IF FNask("Do you wish to format
this disc (Y/N) ?")=TRUE THEN
390 SYS 5,"Format "+STR$drive% +
"D"
400 ELSE
410 IF FNask("Wipe all files on
dest. disc (Y/N) ?")=TRUE THEN
420 SYS "XOS_CLI","WIPE :" +STR$%
drive%+.$.*. FR~C~V"
430 ENDIF
440 ENDDIF
450 IF FNask("Continue (Y/N) ?")=
FALSE THEN ENDPROC
460
470 pre$=:"+STR$drive%+"."
480 *NODIR
490 SYS 5,"DISMOUNT "+STR$drive%
```

```

500 SYS 5,"MOUNT "+STR$drive%
510 FOR ra%=0 TO copycounter%-1
520   spaces%<=2
530   PRINT FNindent;"Dir : "+copylist$(ra%)
540   SYS 8,8,pre$+"$."+copylist$((ra%),0,0,0
550   SYS &29,0,pre$+"$."+copylist$((ra%)
560   PROCexamine(copylist%(ra%))
570 NEXT
580 ENDPROC
590
600 DEF PROCexamine(diraddr%)
610 LOCAL counter%,count%
620 spaces%+=2
630 count%<=0
640 counter%<=diraddr%+5+surface%
650 WHILE ?counter%<>0 AND count%<=77
660   file$=FNmakestring(counter%,10)
670   load%=counter%!10
680   exec%=counter%!14
690   len%=counter%!18
700   sct%=((counter%!22) AND &FFFFF)
710   type%=(counter%?25) AND %1
720 CASE type% OF
730   WHEN 0
740   PRINT FNindent;"Dir : "+file$
750   SYS 8,8,file$,0,0,0
760   SYS &29,0,file$
770   PROCexamine(sct%)
780 OTHERWISE
790   type$=STR$~((load%>>8) AND &FFF)
800   PRINT FNindent;"File: "+file$ +". Type : &" +type$,
810   ,sct%+surface%,sct%+surface%+len%
820 ENDCASE
830 counter%+=26
840 count%+=1
850 ENDWHILE
860 SYS &29,0,"^"
870 spaces%-=2
880 ENDPROC
890
900 DEF PROCfinddirs
910 LOCAL ra%
920 FOR ra%=&C00 TO &C7C00 STEP &400
930   addr%=ra%+surface%
940   IF addr%!1=hugo% THEN
950     directory%(dircounter%)= ra%
960     ra%+=&400
970   directory$(dircounter%) =
980     FNmakestring(addr%+&7DD,10)
990   ENDIF
1000 NEXT
1010 ENDPROC
1020
1030 DEF PROCdisplaydirs
1040 LOCAL ra%
1050 FOR ra%=0 TO dircounter%-1
1060   IF notcopy%(ra%)=FALSE THEN
1070     copylist%(copycounter%)=
1080       copylist$(copycounter%)=
1090         copycounter%+=1
1100   ENDIF
1110 NEXT
1120 ENDPROC
1130
1140 DEF PROCPrompt(text$)
1150 PRINT text$
1160 SYS 6,21,0
1170 IF GET:ENDIF
1180 ENDPROC
1190
1200 DEF FNmakestring(addr%,maxlen%)
1210 LOCAL str$,chr%,cnt%
1220 str$=""
1230 chr%<=65
1240 cnt%<0
1250 WHILE chr%>32 AND chr%<256 AND
1260   cnt%<maxlen%
1270   chr%<=addr%?cnt%
1280   IF chr%>32 AND chr%<256 THEN
1290     str$+=CHR$chr%
1300   =str$
1310
1320 DEF PROCfindroots
1330 LOCAL ra%,ra2%
1340 FOR dir%=0 TO dircounter%-1
1350   taddr%=directory%(dir%)
1360     +surface%
1370   end%<=FALSE
1380   endcounter%<=0
1390   FOR ra%=0 TO 79
1400     addr%=((ra%*26)+5+taddr%)
1410     IF ?addr%<0 AND end%<=FALSE
1420       THEN end%<=TRUE:endcounter%<=ra%
1430     NEXT
1440   IF end%<=FALSE THEN
1450     endcounter%<=ra%-1

```

```

1430 IF endcounter%>0 THEN      1540 DEF FNask(text$)
1440   FOR ra%=0 TO endcounter%-1 1550 PRINT 'text$'
1450     addr%=(ra%*26)+5+taddr% 1560 text$=GET$
1460   FOR ra2%=0 TO dircounter%-1 1570 CASE text$ OF
1470     IF addr%!21=directory% 1580   WHEN "Y","y"
1480       THEN notcopy%(ra2%)=TRUE 1590   =TRUE
1490     NEXT 1600   OTHERWISE
1500   NEXT 1610   =FALSE
1510 ENDIF 1620 ENDCASE
1520 NEXT 1630
1530 ENDPROC 1640 DEF FNindent
                           1650 =STRING$(spaces%,CHR$32[A]

```

First Word Plus Printer Drivers – Part II

Ian Nicholls

Creating Alternative Character Sets

I promised to say something about this last month and why it is relevant to what has gone before. If you have been reading this article with your Archimedes User Guide and printer manuals to hand, you will probably have noticed that there are still quite a few Latin 1 characters that your printer just cannot reproduce. You will also have spotted that there are some characters in your printer's character set that we have not used. You could always take one of these Latin 1 characters that your printer cannot produce, and put against its entry in the Translation Table the code sequence to print one of the so-far unused characters in your printer's character set. The problem with this is that you will have to remember which character in the FWP on-screen font table corresponds to the character you want your printer to produce. What you really need is to be able to alter the character depicted in the font table to look like the one the printer will produce.

If you are an old BBC micro hand, or you are an avid reader of your User Guide, you will know about the VDU 23 command. This command enables you to redefine the shape of any of the printable characters: if you turn to pages 50 and 51 of the User Guide you will see how to make ASCII code 128 print a nice little space invader. Once you have mastered creating your own character definitions with VDU 23, you need to define the sets of eight bytes which would draw each of the unused characters in your printer's character set. The last trick is how to redefine these characters automatically when loading FWP.

In the following program I have redefined quite a large number of the Latin 1 characters that my printer cannot print into characters that it can print (but which are not in Latin 1). The lines that perform these redefinitions are the DATA statements in lines 230 to 491. The program creates a command file using the SPOOL command in line 50: the file is called "gem_codes". Lines 60 to 140 read the DATA statements and turn them into the corresponding VDU 23 commands. When the statement beginning with 999 is reached the command *SPOOL is issued to close the command file that has been created.

```

10 REM > VDU_23
20 RESTORE
30 DIM A%(7)
40 MODE0
50 *SPOOL gem_codes
60 WHILE FNread_ascii_code<>999
70   FOR J%=0TO7
80     READ A%(J%)
90   NEXT
100  VDU 23,code%
110  FOR J%=0TO7
120  VDU A%(J%)
130 NEXT
140 ENDWHILE
141 *SPOOL
150 VDU19,7,0|19,0,7|
160 FOR I%=160TO255
170   PRINT CHR$(I%); " ";
180 NEXT
190 END
200 DEFFNread_ascii_code
210 READ code%
220 =code%

```

```

230 DATA 160,24,60,126,24,24,0,0,0
240 DATA 164,0,0,24,24,126,60,24,0
250 DATA 171,0,12,60,255,60,12,0,0
260 DATA 172,0,48,60,255,60,48,0,0
270 DATA 173,0,28,99,99,99,28,0,0
280 DATA 174,28,54,127,0,0,0,0,0
290 DATA 175,127,54,28,0,0,0,0,0
300 DATA 178,192,240,204,240,192,0,0,0
310 DATA 179,3,15,51,15,3,0,0,0
320 DATA 180,24,102,195,102,24,0,0,0
330 DATA 182,24,126,255,126,24,0,0,0
340 DATA 185,254,198,198,198,254,0,0,0
350 DATA 187,252,48,48,53,50,53,0,0
360 DATA 190,24,24,126,219,126,24,24,0
370 DATA 192,60,102,102,126,102,102,
       60,0
380 DATA 193,112,112,15,24,24,24,15,0
390 DATA 194,112,112,15,24,31,24,24,0
400 DATA 199,0,62,99,99,99,99,54,99,0
410 DATA 200,0,99,54,99,99,99,62,0
420 DATA 202,0,127,48,24,24,48,127,0
430 DATA 203,0,15,60,102,102,102,60,0
440 DATA 204,0,127,54,54,54,54,54,0
450 DATA 205,62,99,3,3,3,99,62,0
460 DATA 206,31,48,254,48,126,219,
       243,0
470 DATA 207,127,0,60,6,58,102,63,0
480 DATA 208,252,204,252,0,0,0,0,0
490 DATA 210,24,24,126,24,24,24,24,24,0
491 DATA 195,0,126,0,60,102,126,102,0
500 DATA 999

```

If this program is now run, the file called "gem_codes" will be saved on disc and the complete character set of ASCII codes from 160 to 255, including the redefined characters, will be displayed on screen. All that is needed to load these redefined characters into the Archimedes is to issue the command:

```
*PRINT gem_codes.
```

To make FWP load this file automatically, you first need to save it on your FWP system disc. I assume that you have followed Acorn's advice, and you have copied the FWP system disc before starting to create and store your own documents. I will also assume that you have run the above program and that you now have the file "gem_codes" in the root directory on a work disc. The first step is to copy the file "gem_codes" onto your work copy of the FWP system disc (not onto the master system disc). The

appropriate place to put "gem_codes" is in the Library directory. If you have an Archimedes with one floppy disc installed, the command to issue is as follows:-

```
*COPY :0.$.gem_codes :0.$.Library
      .gem_codes P Q
```

You should place the work disc with "gem_codes" on it into the disc drive when asked to put the source disc in, and then replace it with the working copy of the FWP system disc when asked to insert the destination disc.

The last step is to alter the "boot" file which the Archimedes reads from the FWP system disc when you press Shift/Break. The boot file has the name "!BOOT" and is, in fact, a four line BASIC program which is run when you press Shift/Break; the program is:-

```

10 REM > !boot for First Word Plus
20 *FX225,1
30 *key 1 QUITM1stWord+M
40 *FX138,0,129

```

Insert the work copy of the FWP system disc into the drive and type LOAD "!BOOT": then type LIST and the above program will be listed. Now type in this extra line:-

```
15 *PRINT $.Library.gem_codes
```

You should save the modified boot program by typing SAVE "!BOOT". To help you follow the article more easily, two printer driver hex files have been included on this month's program disc for the Star Gemini 10X printer (star-gem1 and star-gem2): the corresponding configuration files (star-stdrd and star-nlq) are also included. In addition, the "VDU_23" program and its output file "gem_codes" are on the program disc. The two printer driver files are almost the same; however, one of them "star-stdrd" prints normal text in the standard Pica typeface, the other, "star-nlq", prints normal text with the emphasised Pica typeface to give a fuller character. Hopefully, this article will enable you to write your own printer drivers and to make full use in FWP of your printer's complete range of characters. A

Assembly Language Programming – 6

Alan Glover

In this month's part we are going to examine all the ARM instructions which fall in groups 1 and 1A – which includes all the arithmetic and logical operations. This is big chunk so there's no space for a demo program this time. Sorry !

Group 1 – Logical Instructions

This sub-group all have the same effect on the flags when they are told to amend them (by adding an 'S' to the opcode). The N bit is set to the value of bit 31 of the result. The Z flag is set if the result is zero, and cleared otherwise. The C flag is only affected by the <right hand side> shifts mentioned previously. The V flag is not affected.

AND(<conditional>)(S) <dest>,<lhs>,<rhs>

Destination register is set to <lhs> AND <rhs>, e.g.

```
ANDS R0,R1,#&20000000 (clear all bits  
except bit 29)
```

```
ANDS R0,R1,R2 (result is R1 AND R2)
```

```
ANDS R0,R1,R2,LSL #4 (R2 is shifted left four  
times, then ANDed with R1 to produce a  
result in R0. Note – R2 is not changed by  
this.)
```

BIC(<conditional>)(S) <dest>,<lhs>,<rhs>

BIC represents BIte Clear. It ANDs the left hand side with the inverse of the right hand side. This is useful if you want to reset just one bit, e.g.

```
BIC R0,R0,#&20000000 (reset bit 29 of R0)
```

TST(<conditional>)(P)(S) <lhs> <rhs>

This instruction TeSTS the lhs and rhs by ANDing them. The 'S' is always assumed and need not be explicitly stated. If a 'P' is added bits 31-28 of the result will be set as the current status flags. This is useful when the flags are being directly manipulated, e.g.

```
TST R0,#&20 (set Z flag by whether bit 5 is 1  
or 0)
```

ORR(<conditional>)(S) <dest> <lhs> <rhs>

This instruction ORs the lhs and rhs, e.g.

```
ORR R0,R0,#&20 (set bit 5 in R0)
```

EOR(<conditional>)(S) <dest> <lhs> <rhs>

This instruction Exclusively ORs the lhs and rhs. An exclusive OR will result in 1 if the two bits being compared are different, and 0 if they are the same. This is useful for 'toggling' bits – alternately setting them to 0 and 1, e.g.

```
EOR R0,R0,#1 (change bit 0 of R0)
```

TEQ(<conditional>)(P)(S) <lhs> <rhs>

This instruction Tests for EQUALity by exclusively ORing the lhs and rhs. If they are the same this will result in zero. This has the advantage over a CMP or CMN instruction that only the Z flag will change, e.g.

```
TEQ R0,#&45 (set Z flag if R0=&45, otherwise  
clear it)
```

The 'S' is automatically assumed and can be omitted. The 'P' option is used to write bits 31 to 28 of the result onto the status flags.

The example below is one way of setting the overflow flag:

```
MOVVC R0,R15 (move R15 to R0)  
TEQVCP R0,#&20000000
```

The rules governing the use of R15 are explained later, and explain why R15 could not be used directly in the TEQ instruction.

MOV(<conditional>)(S) <dest> <rhs>

This instruction is used for setting one register to the value given immediately or held in another. Note that the second parameter is a rhs, so the full range of shifts is available, e.g.

```
MOV R0,R0,LSL #8 (multiply by 256, 2 to the  
power of 8), or
```

```
MOV R0,#0 (setting a register to 0)
```

```
MOV R1,R0 (loading R1 with the value of R0)
```

MVN(<conditional>)(S) <dest> <rhs>

This is like MOV, but stands for MoVe Negated. The rhs is inverted before use (after any shifts etc have taken place).

One of the quirks of signed notation against simple inversion is that the results are not the same.

Inverting 1 produces &..FE, which is -2. Whereas in signed notation the result is &..FF.

Typically you use one less than you want, e.g.

MVN R0, #0 (load R0 with -1, &FFFFFF)

Group 1 – Arithmetic Instructions

These instructions still set the N and Z as detailed above, but will also set C if a carry occurs from bit 31, and V if a carry occurs from bit 30.

ADD(<conditional>)(S) <dest> <lhs> <rhs>

This instruction adds the lhs and rhs. It should be used as the first instruction of a multi-word addition with the 'S' present.

ADC(<conditional>)(S) <dest> <lhs> <rhs>

This instruction adds the lhs, rhs and carry flag. The carry flag is used to convey the need to carry in from the previous addition, e.g.

ADDS R0, R0, R4 (R0=R0 + R4, carry flag set by result)

ADCS R1, R1, R5 (R1=R1 + R5 + c, carry flag used, and set by result)

ADC R2, R2, R6 (R2=R2 + R6 + c, carry flag used, but not changed).

If you wanted to test for overflow of the addition after adding R2 and R6 you would include 'S' on the final ADC, and use conditional execution to deal with it appropriately, e.g.

ADCS R2, R2, R6

BCS overflowed (Branch (explained later) to 'overflowed' if C set)

SUB(<conditional>)(S) <dest> <lhs> <rhs>

Subtracts the rhs from the lhs.

SBC(<conditional>)(S) <dest> <lhs> <rhs>

Subtracts the rhs and the inverse of the carry flag from the lhs. This instruction would be used on the second and subsequent operations of a multi-byte subtraction. The usage is the same as the ADD/ADC example above.

RSB(<conditional>)(S) <dest> <lhs> <rhs>

Performs a Reverse Subtraction, i.e. the lhs is subtracted from the rhs. This allows the additional power of rhs operands to be used on the other side of the subtraction. This is necessary because

subtraction is not commutative ($2+3 = 3+2$, but $2-3 \neq 3-2$).

RSC(<condition>)(S) <dest> <lhs> <rhs>

This is another Reverse Subtraction, but using the Carry flag too, hence it is analogous to SBC.

CMP(<condition>)(P)(S) <lhs> <rhs>

This instruction compares two numbers by subtracting the rhs from the lhs. As with TEQ and TST the 'S' is always assumed present, and 'P' may be used to write to the status bits in R15 from bits 31 to 29 of the result.

CMN(<condition>)(P)(S) <lhs> <rhs>

This is a way of comparing negative numbers. Unlike MVN no adjustment is needed to the number, e.g.

CMN R0, #1 (tests whether R0 = -1, NOT -2 as it would be in MVN)

Group 1 – Additional Instructions

These three instructions are closest to group 1, so are being included here.

ADR(<condition>) <dest> <address>

Used to load a register with the address of certain point, but derived in a relative form which works wherever the program is loaded (provided that the point being referenced has maintained its offset from the ADR instruction). In fact this is not a true instruction, but a pseudo instruction provided by the Basic Assembler. It is in fact an ADD or SUB instruction depending where the label is, e.g.

ADR R0, message (load R0 with the address of 'message')

MUL(<condition>)(S) <dest> <lhs> <rhs>

This instruction multiplies the lhs by the rhs. The N and Z flags are set normally, but V is unchanged, and C is undefined. The dest and rhs must be different registers. The range of rhs options is slightly restricted – shift #n cannot be used, though shift Rn can.

MLA(<condition>)(S) <dest> <lhs> <rhs> <add>

This is another multiplication instruction. After multiplying the lhs and rhs the add is added on to the result. The restrictions noted above apply here too.

Group 1 – Using R15

If R15 is used as a lhs only the PC portion (bits 2-25) will be seen as the lhs, however all bits appear when it is used as a rhs.

If TEQ/TST/CMN/CMP are being used with 'P', bits 31-28 of the result will overwrite the status flags allowing direct alteration.

For the other instructions, if 'S' is absent then only the PC portion will be altered.

When the 'S' is present all bits are altered. N.B. the status bits will be bits 31 to 28 of the result, not the status flags at the end of the operation.

R15 should be manipulated with care. Because of pipe-lining it is usually 8 bytes ahead of the instruction being executed, so trying to alter R15 directly needs care!

Next month we look at groups 2 and 3. **A**

A Music Keyboard for Archimedes

Ronald Alpiar

I suppose that a large proportion of Archimedes owners will have graduated from Acorn's BBC micros, and of these many may have purchased musical peripherals, that is, ATPL's 'Symphony 500' keyboard, with Hybrid Technology's 'Music 500/5000' synthesiser. Quite splendid results can be obtained using this setup, together with the AMPLE programming language supplied with the synthesiser. The system is far more versatile than a cursory examination of the AMPLE manuals indicates, and is capable of supporting a wide variety of experimental music, and generating beautiful stereo sounds.

Like me, these users will have been asking themselves whether these same peripherals can be physically and functionally connected to Archimedes. After all, they are only connected to the BBC's User Port, and 1 MHz bus respectively, both of which are available on an installed Archimedes I/O Podule. As so often, the answer is both 'yes' and 'no'!

First, the bad news. Although Hybrid's synthesiser can easily be plugged into the I/O Podule's 1 MHz bus, we still need to know the message protocol, and how the synthesiser interprets instructions – that is, how to program the synthesiser. Hybrid have been approached with a request for this data. Unfortunately this is regarded as a commercial secret, and Hybrid's firm reply implies that I would not be allowed to publish this information, even if I had found it out – which I haven't!

However, all is not lost. The good news is that ATPL's keyboard can easily be plugged directly

into the Podule's User Port, and that a simple BASIC program suffices to handle it. The Archimedes audio hardware generates sounds which (though far below the standard of Hybrid's synthesiser) are acceptable, especially if heard via a decent stereo system connected to Archimedes' audio socket. By this means, full 8-note polyphony is attainable: moreover Ian Nicholls' Soundsynth programs (Archive Vol.1, Nos.8-10) and EMR's SoundSynth software can be used to expand Archimedes' somewhat limited repertoire of built-in waveforms.

The ATPL 'Symphony 500' Keyboard

We have here a 4-octave, full sized keyboard, including a pedal (sustain) socket – though alas, understandably in view of its very modest price, not offering touch sensitivity. Electronics within the keyboard unit utilises the User Port's 1 MHz clock signal and others to transmit a continuous stream of bytes to the User Port. Each bit of a byte will be zero or 1 according as to whether the corresponding key is depressed. Thus byte 1 maps the status of keys 1-8, byte 2 keys 9-16 etc. ATPL provide machine code software which sits in lower RAM of the BBC, scans the incoming byte stream, and assigns and releases sound channels as required. Further software in an higher language (BASIC or AMPLE) can then access this information and issue appropriate sound commands to the output device (which may be either the internal BBC sound chip, or Hybrid's synthesiser).

Since a User Port is needed, the Archimedes must be equipped with an I/O Podule. We all know how much faster Archimedes BASIC is compared to BBC's, but is it fast enough to scan the keyboard

and generate up to 8-voice polyphony, or must we (as with the BBC) resort to machine code? In other words, can BASIC keep up with the fastest feasible playing speeds without missing notes? The answer is yes – but only just!

The BASIC program – how it works

The program consists of a main section which first sets up initial values (PROCsetup) and then cycles continuously from reading the keyboard (PROCread_key) to producing the required sounds (PROCplay_on).

PROCsetup

An understanding of the arrays and constants defined here will make the explanation of the subsequent procedures much clearer. They are:-

nb is an array of 8 bytes. Each byte has 1 in all but one of its 8 bits: thus $nb?0=%11111110$, $nb?1=%11111101$ etc. **nb** is used to send control information to one of the registers of the VIA (6522) chip, which controls the User Port pins.

key is an array of 48 bytes, one for each key of the keyboard. If a given key is free, the corresponding byte is zero: if depressed then the byte is non-zero (its actual value doesn't matter).

cmap is an array of 8 bytes, one for each of Archimedes' maximum 8 voice channels. If a channel is currently idle, then the corresponding byte is zero: when busy, the byte=1.

kmap is another 48 byte array for the keyboard. If a given key is depressed and has been assigned channel #n then the value of the byte=n : otherwise its value=0.

pmap % is an integer array which maps each key of the keyboard to the corresponding Archimedes fine pitch (User Guide page 185).

Certain default variables are set up in line 170. 'transpose%' sets the overall tuning, such that the middle note of the keyboard plays middle C. 'nchan%' sets the number of channels, which determines how many notes can play simultaneously. 'nnotes%' tells the program how many notes of the keyboard to scan, and, pretty obviously, 'volume%' sets the volume of the resulting sounds. 'voice%' tells which software voice is being chosen to express the sounds. Voice

#2 is 'Stringlib-soft' included in Arthur. A set of OSCLI calls effect *CHANNELVOICE commands.

Before ENDPROC, 'setup' issues two crucial commands to the VIA (lines 270, 280). First to tell it that all 8 pins of the User Port are to be treated as inputs: second, to set the bits in the VIA's 'Auxiliary Control Register' to %00011000 : this ensures that the shift register feeds the data out under control of the 1 MHz clock pulse. (Actually the Archimedes I/O Podule operates a clock at 2 MHz; how fortunate that ATPL's keyboard has no difficulty in keeping up!)

PROCread_keyboard

This procedure reads a series of 6 bytes from the User Port. Each byte represents the status of 8 consecutive keyboard keys. Question: How do we know just which octet of keys corresponds to a given byte? The answer lies in line 340, which sets the corresponding bit of VIA's Shift Register low, just before the User Port pins are read (line 350). Thereafter all we need do us to 'unpack' the bits of each byte into successive positions of the array 'key' (we shall explain why lines 370-440 are so clumsily programmed a little later).

PROCplay_on

The status of each key in turn is examined by looking at the value of the byte array 'key'. This is compared with the channel status of that key in array 'kmap'.

If a key which was previously released is now depressed (offon%>0 AND chan%=0) then a free channel is sought (if none is found the key depression remains inactive till the next scan cycle). When a free channel is found it is grabbed (set cmap) and the kmap byte is set to show which channel is being used. Lastly a SOUND instruction is issued.

The reverse actions take place if a key previously depressed is not released. The channel is freed, and the corresponding 'kmap' byte=0, and finally the sound is terminated.

FNfind_free_channel

PROCplay_on uses this function which returns the channel number (>0) of the first idle channel, or returns zero if all channels are busy.

Timing Considerations

A complete PROCread_keyboard : PROCplay_on cycle takes 30.3 milliseconds.

By comparison, fairly agile fingers can execute one of the fastest musical actions, a trill, at the rate of about 16 note-changes per second. So new notes have to be sounded every 62.4 milliseconds. If we allow for daylight gaps between the notes, to give a trill its proper clarity, we must allow for a change in keyboard status every 31.25 milliseconds.

In order to maximise execution speed, shaving away every surplus millisecond, the program incorporates a few compromises. For instance, neither the top (49th) note nor the pedal switch of the keyboard are scanned. This means that the FOR loop in line 330 is limited to x%=5 rather than x%=7. And lines 370, 440 mysteriously execute faster when written out, compared to sitting in a FOR loop.

Modifications and extensions

The user may alter the settings of the default parameters (line 170) with interesting results. It would also be possible to assign a different voice to each channel – not a lot of use since as written PROCfind_free_channel merely returns the number of the first free channel, irrespective of other considerations. But in principle it would be possible to partition the keyboard over several different voices, or to perform quick voice changes using the pedal key. A more refined FFind_free_channel would release the longest occupied channel if a new channel was needed at a time when all were busy. Needless to say, all these refinements would carry a timing penalty!

More interestingly, one might experiment with line 210 which sets the pmap% array. As it stands, this is set to generate the modern 12-note equally tempered scale. It would be very easy to alter the formula to generate say a 31-tone equal tempered scale (following a school of sound pioneered in Holland, which can produce some of the most rapturous and un-earthly music I've heard). Alternatively we might experiment with some of the precursors of modern musical temperament – the pentatonic, pythagorean, just, or mean tempered scales: it would be quite easy to effect the long

sought-after musical miracle of an instrument which plays a completely ‘correct’ scale corresponding to the key of the piece of music being performed.

Postscript

The above program is an offshoot of the author's principal present preoccupation. This is to write a program ('MUSICK') which will enable the user to perform prescribed pieces of music, without playing the actual notes – much as an orchestral conductor. He is given complete freedom over expression, dynamics, phrasing, use of glissando vibrato and other effects in real time during performance and is thus able to express musicianship without the slog of learning to play an instrument – though to be fair – I have really had the disabled in mind when doing this work. MUSICK is now at a very advanced stage: when complete, it will be advertised (like my Astronomy program SKY-BABY) as a free gift to raise funds for charity. Watch this space!

```
10 REM Program to play from ATPL's
      keyboard
20 REM connected to User Port of I/O
      Podule
30 PROCsetup
40 REPEAT
50  PROCread_keyboard
60  PROCplay_on
70 UNTIL FALSE
80 END
90
100 DEF PROCsetup
110 DIM nb 7, key 63, cmap 7, kmap
      63, pmap%(63)
120 z%=1
130 FOR i%=0 TO 7
140   nb?i% = NOT z%
150   z%=2*z%
160 NEXT
170 transpose%=24 : nchan%=4 : voice%
      =2 : nnotes%=48 : volume%=-15
180 FOR i%=0 TO 7 : cmap?i%=0 : NEXT
190 FOR i%=0 TO 63
200   kmap?i%=0
210   pmap%(i%)=&4000 + ((&1000*(i-
      transpose%))DIV12)
220 NEXT
230 VOICES nchan%
```

```

240 FOR i%=1 TO nchan%
250   OSCLI "CHANNELVOICE " + STR$(i%) + " " + STR$(voice%)
260 NEXT
270 SYS 6,151,&62,0
280 SYS 6,151,&6B,&18
290 ENDPROC
300
310 DEF PROCread_keyboard
320 pos%=key
330 FOR x%=0 TO 5
340   SYS 6,151,&6A,nb:x%
350   SYS 6,150,&60 TO ,,port%
360   nport%=NOT port%
370   ?pos%=nport% AND 1:pos%+=1
380   ?pos%=nport% AND 2:pos%+=1
390   ?pos%=nport% AND 4:pos%+=1
400   ?pos%=nport% AND 8:pos%+=1
410   ?pos%=nport% AND 16:pos%+=1
420   ?pos%=nport% AND 32:pos%+=1
430   ?pos%=nport% AND 64:pos%+=1
440   ?pos%=nport% AND 128:pos%+=1
450 NEXT x%
460 ENDPROC
470
480 DEF FNfind_free_channel
490 FOR i%=0 TO nchan%-1
500   IF cmap?i%=0 THEN =i%+1
510 NEXT i%
520 =0
530
540 DEF PROCplay_on
550 FOR n%=0 TO nnnotes%-1
560   offon%=key?n% : chan%=kmap?n%
570   IF offon%=0 AND chan%==0 GOTO 710
580   IF offon%>0 AND chan%==0 THEN
590     c%=FNfind_free_channel
600   IF c%==0 ENDPROC
610   kmap?n% = c%
620   cmap?(c%-1) = 1
630   SOUND c%,volume%,pmap%(n%),255
640   GOTO 710
650 ENDIF
660   IF offon%==0 AND chan%>0 THEN
670     cmap?(chan%-1)=0
680     kmap?n% = 0
690     SOUND chan%,0,0,0
700   ENDIF
710 NEXT n%
720 ENDPROC A

```

Thoughts of a PC Emulator User

Brian Cowan

Even if I say so myself, one quality that I possess (at least as far as computers goes) is that of loyalty. Of course this may also be interpreted as conservatism or laziness. Thus I have never used a PC or even a Mac. It should be said that I was a slow convert to the old BBCs, having bought my first one to use as a terminal for a mainframe computer. My conservatism caught me out here also however; I preferred the old dedicated terminals so the Beeb was hardly used!!.

Conversion to BBC

One day I had some calculations to do which were a little too complex for my pocket calculator, but too simple for a mainframe, and the RML 380Z which we were using at that time was in use. An ideal opportunity for the BBC. And once bitten I was hooked. My initial objection was based on the view of the 6502 as a prehistoric processor. It was

experience of the firmware user interface; the operating system and the BASIC, which converted me. This provided versatility and power and simplicity. Thus I became hooked.

So why the reminiscences and what relevance is all this to the Archimedes PC emulator? The question is: where does loyalty end and simple practical considerations take over? Our esteemed editor Paul B. has one view. As is well known, Archive magazine is produced on a Mac. (Soon to be swapped for a Mac II to speed up the production process! Ed.) It so happens that the Mac has what is probably the best desk top publishing system of any micro. But my requirements are somewhat different. I need to do number crunching and word processing and controlling scientific instruments including taking experimental data and real time data processing. The Archimedes is ideal for performing all these tasks.

The value of PCs

However one can't ignore the vast array of software for the IBM PC and its clones. Acorn certainly didn't and the PC emulator was the result. In this way the world of PC software is available to the Archimedes user. I have recently been making use of this facility, using PC packages for performing computer algebra. I started doing computer algebra on my old Beeb, writing the programs in BBC BASIC. Unfortunately I soon came up against speed and memory problems. The solution then was to use the Acorn 32016 second processor running programs overnight! I recently returned to some of these problems and started running the old programs on the Archimedes. This was also outgrown and it was obvious that I should resort to commercial packages.

Hardly surprisingly, most software was for PCs and their clones. Accordingly I started using a system called Mumath. This was most successful, but again we soon exhausted its capabilities and a more sophisticated tool was required. We graduated to Reduce which was originally written for mainframes. This was superb except for the problem of speed. At this stage I would not have known any better had Acorn not made the next move.

Acorn Master 512

The price of the 512 PC board for the Master was reduced to ninety nine pounds. At this price who could resist? – Recall that it started life costing four hundred pounds. I purchased one of these coprocessors, dug out my old Master and away we went. Imagine my horror when I discovered that the Mumath programs zapped along very much faster than on my Archimedes! I was in a quandary. Perhaps I should go the whole hog, buy a real PC and be damned.

PC Emulation

I am not belittling the PC emulator for one moment; I think this is a brilliant piece of programming. Also, I know that as later and later versions are issued they run faster and faster. However there is clearly a limit, and a little simple mathematics should tell what is the maximum realisable speed of a software

emulator, based on the Archimedes clock speed and the average number of RISC instructions to make a PC instruction. It is my opinion that there is no further significant speed increase to be made, although I know that journalists in some of the other computer press think otherwise.

A PC Podule?

The PC emulator is perfectly adequate for running the occasional MSDOS program, and of course that was all it was intended for. Serious MSDOS users should presumably use a PC, but what about the occasional serious user? The answer would be a PC podule with a PC processor but, as we all know, Acorn have discontinued the PC podule project. This was done for two reasons. Firstly with improved versions of the emulator its compatibility has reached startling limits – much greater I believe than was ever expected, and one of the reasons for a hardware emulation was the compatibility issue.

The second reason for the demise of the PC podule was the evolution of later Intel processors. The PC podule, as the Master 512 board, was to be based on the 80186 chip. But this is already old hat. We have the 80286 and the latest machines are using the 80386. In developing a 186 board Acorn could be regarded as flogging a dead horse, particularly considering the development costs and their reflection in the final price of the podule.

Thoughts for the future

Maybe Acorn will develop a podule based on these later chips. I think not. The market here will probably be satisfied by third party sources. I hope some people are getting on with this. One possible solution would be to develop a simple podule which would take a Master 512 board. I think this deserves serious consideration. This could possibly be done in conjunction with using the 6502 emulator for some of the I/O. However, this may prove to be an impossibly crazy suggestion. (I have just discovered that this has been done already within Acorn, but not for public release – they produced a Tube podule.) What is clear is that a hardware PC emulator would be a popular addition to the Archimedes range of add-ons. So come on then..... A

Fact-File

ABACUS Training	29 Okus Grove, Upper Stratton, Swindon, Wilts, SN2 6QA.
ACE Computing	27 Victoria Road, Cambridge, CB4 3BW. (0223-322559)
Akom Ltd	51 Manor Lane, London, SE13 5QW. (01-852-4575)
Alpine Software	P.O.Box 25, Portadown, Craigavon, BT63 5UT. (0762-42510)
Beard Technology	111 Evering Road, London, N16 7SL.
Brainsoft	22 Baker Street, London, W1M 1DF. (01-486-0321)
CJE Micros	78 Brighton Road, Worthing, W Sussex, BN11 2EN. (0903-213361)
Clares Micro Supplies	98 Middlewich Road, Rudheath, Northwich, Cheshire, CW9 7DA. (0606-48511)
Colton Software	149-151 St Neots Road, Hardwick, Cambridge, CB3 7QJ. (0954-211472)
Computer Concepts	Gaddesden Place, Hemel Hempstead, Herts, HP2 6EX. (0442-63933)
Computerware	55 Romulus Court, Brentford Dock, Justin Close, Brentford, Middlesex.
Contex Computing	15 Woodlands Close, Cople, Bedford, MK44 3UE. (02303-347)
Dabs Press	76 Gardner Road, Prestwich, Manchester, M25 7HU. (061-766-8423)
GEM Electronics	17 Tandragee Road, Portadown, Craigavon, BT62 3BQ.
Ian Copestake Software	10 Frost Drive, Wirral, L61 4XL. (051-648-6287)
IFEL	36 Upland Drive, Plymouth, Devon, PL6 6BD. (0752-847286)
Intelligent Interfaces	43B Wood Street, Stratford-on-Avon, CV37 6JQ. (0789-415875)
Impact	Unit 3, 1 Percy Street, Sheffield, S3 8AU.
Minerva Systems	69 Sidwell Street, Exeter, EX4 6PH. (0392-437756)
Pineapple Software	39 Brownlea Gardens, Seven Kings, Ilford, Essex, IG3 9NL. (01-599-1476)
P.R.E.S.	6 Ava House, Chobham, Surrey. (0276-72046)
Psientific Software	37 Cotesmore Road, Hessle, North Humberside, HU13 9JQ. (0482-649187)
Richard Millican	10 Stokesay, Bidston, Birkenhead, L43 7PU
Silicon Vision Ltd	Signal House, Lyon Road, Harrow, Middlesex, HA1 2AG. (01-422-2274)
Solidisk Technology	87 Bournemouth Park Road, Southend-on-Sea, SS2 5JJ.
Superior Software	Regent House, Skinner Lane, Leeds, LS7 1AX. (0532-459453)
Video Electronics Ltd	Wigan Road, Atherton, Manchester, M29 0RH.
Z & Z Software	Brecklands, Broad Oak, Shrewsbury, SY4 3AH.

Archive

Subscription Magazine and Support Group for *Archimedes* Users

Archive Magazine contains:

- News
- Reviews
- Hints and Tips – a major feature
- Articles for Beginners
- The Latest Technical Information
- Program Listings
- Free Small Ad's Section
- HELP – Requested and Offered
- Contact Box – to help you form common interest groups

Eureka! – Bulletin Board

0603-250689 on 1200/75 or 300/300

- Very Latest News
- Down-load Software
- Mailbox Facilities
- Chat line

N.B. All earlier issues have now been re-printed – you may back-date your subscription as far as issue 1 (October 1987) – to take advantage of this huge bank of information.

Archimedes is a trademark of Acorn Computers Ltd.

* Please send copies of *Archive* magazine for one year starting from
Volume ____ Issue ____

* Please enrol me on the Technical Help Service for one year. (£8)
I enclose a cheque for £ _____ payable to "Norwich Computer Services".

Name: _____

Address: _____

Postcode: _____

Norwich Computer Services, 18 Mile End Road, Norwich, NR4 7QY

Special two-year subscription starting from volume 1, issue 1 – £27